DOCUMENT RESUME

ED 335 220 SE 052 187

AUTHOR Schuman, Jonathan C.
TITLE Teaching Two-Step Numerical Problem Solving to Elementary School Students through Computer-Assisted Instruction.
PUB DATE Mar 91
NOTE 121p.; M.S. Practicum Report, Nova University.
PUB TYPE Dissertations/Theses - Practicum Papers (043) -- Tests/Evaluation Instruments (160)

EDRS PRICE MF01/PC05 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; *Elementary School Mathematics; Grade 2; Grade 3; Individualized Instruction; *Learning Disabilities; *Learning Modalities; Mathematics Education; *Mathematics Instruction; Perceptual Handicaps; Pretests Posttests; Primary Education; Problem Solving; Programed Instruction; *Small Group Instruction
IDENTIFIERS Mathematics Education Research

ABSTRACT

Previous research has indicated that numerically based problem-solving can be taught to children and that it should be initiated on a continuing basis in the elementary school years, yet there is no evidence for the existence of an exact, comprehensive process through which problem-solving with numerical comparisons and addition or subtraction operations can be effectively taught to children. The ability of 6 second- and third-grade students with various learning difficulties to use problem-solving skills in making decisions about open arithmetic sentence comparisons through computer-assisted instruction was examined during a 6-week summer program. Within self-paced environments monitored by a graduate student, pairs of students made independent use of various computer softwares modified to suit individual needs across several learning modes. The results indicated that single-step problem solving can be taught in a relatively short period of time using computer assisted instructional strategies. However, complex problem solving, which involved critical thinking and multiple tasks beyond mere recognition and memorization, was not fully attainable by second- and third-grade students with learning difficulties, particularly within such a short learning period. (Appendices include 22 references, pretests, posttests, student instructional preference survey, software recommendation sheet, and student data in the form of a daily log.) (JJK)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED335220

TEACHING TWO-STEP NUMERICAL PROBLEM SOLVING TO ELEMENTARY SCHOOL STUDENTS THROUGH COMPUTER-ASSISTED INSTRUCTION

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☒ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Jonathan Schuman

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

by

JONATHAN C. SCHUMAN

A Practicum Report
submitted to the Faculty of the Center for the
Advancement of Education of Nova University in partial
fulfillment of the requirements for the degree of
Master of Science.

The abstract of this report may be placed in a
National Database System for reference.

March/1991

BEST COPY AVAILABLE

SE 052 187

Abstract

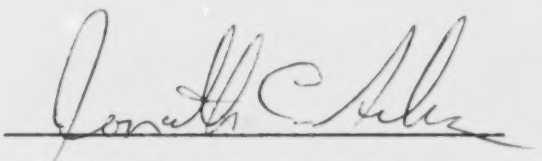
Teaching Two-Step Numerical Problem Solving to Elementary School Students Through Computer-Assisted Instruction.

Schuman, Jonathan C., 1991: Practicum Report, Nova University, Center for the Advancement of Education. Descriptors: Computer Uses in Education/Computer-Assisted Instruction/Computer-Assisted Learning/Elementary Education/Mathematical Skills/Learning Disabilities/Learning Modalities/Perceptual Handicaps/Attention Deficit Disorders/

The improvement of second and third-grade students' ability to use problem-solving skills in making decisions about open sentence comparisons through computer-assisted learning was examined. The students had various learning difficulties and were enrolled in a summer program to improve their skills. The author implemented several instructional strategies and philosophies espoused by Costa (1985) concerning student computer-teacher interactions. Students were placed in a nonthreatening self-paced environment in class sizes of two each with independent use of a computer. The period of implementation, only six weeks, was a short period causing the overall program to be compressed in structure and selective in outcomes. The author appealed to several learning modes simultaneously as proposed by Brevil (1987) while modifying software to meet individual student needs (Eiser, 1986). The students were first introduced to comparative operations with numbers less than 10. Gradually, they worked up to two-step open sentence comparisons requiring addition and/or subtraction of numbers up to 23. The results indicated that single-step problem solving can be taught in a relatively short time using computer-assisted instructional strategies. However, complex problem solving involving critical thinking and requiring more than recognition and memorization were not fully attainable by second and third-grade students with learning difficulties in the short implementation period. The author called for additional research on this subject utilizing a larger target group with increased student-computer contact time. The author also recommended that computer science teachers assume a leadership role in their schools to facilitate the development of problem-solving skills through all subject areas with increased student-computer software interaction. Appendices include pretests, posttests, student instructional preference survey, software recommendation sheet, and student data in the form of a daily log.

Authorship Statement

I hereby testify that this paper and the work it reports are entirely my own. Where it has been necessary to draw from the work of others, published or unpublished, I have acknowledged such work in accordance with accepted scholarly and editorial practice. I give this testimony freely, out of respect for the scholarship of other workers in the field and in the hope that my own work, presented here, will earn similar respect.



Jonathan Craig Schuman

Table of Contents

	Page
Title Page	i
Abstract	ii
Authorship Statement	iii
Observer's Verification	iv
Table of Contents	v
List of Tables	vii
Chapters	
I. Purpose	1
II. Research and Solution Strategy	11
III. Method	22
IV. Results	37
V. Recommendations	44
Reference List	47
Appendices	49
Appendix A: Student Skill Development Logs	50
Appendix B: Target Group Identification Evaluation	88
Appendix C: Skill Assessment Pretest	90
Appendix D: Sign Identification Posttest	95
Appendix E: Student Preference Survey	97

Page

Appendix F: Computer Practicum Contract 99

Appendix G: Creative Student Work 102

Appendix H: Number Value Comparison Pretest/Posttest 105

Appendix I: Comparing Simple Open Mathematical Sentences Pretest/Posttest 109

Appendix J: Instructional Preference Survey 111

Appendix K: Software Recommendation Sheet ... 113

List of Tables

		Page
Table 1:	Identifying Less Than, Greater Than, and Equal to Symbols - Posttest Scores	38
Table 2:	Number Comparison (0-100)	39
Table 3:	Comparing Open Sentences of Addition and Subtraction	41
Table 4:	Instructional Methods: Student Preferences	42

CHAPTER I

Purpose

This practicum took place at an independent elementary school in a large metropolitan area. The 52 predominantly White students involved in the six-week summer program came from upper-middle income homes within the mixed socio-economic setting of the area. The reason for the unusual economic grouping is that the students paid a fee to enter the program. Scholarships were available, but limited. Further examination of the students indicated that some exhibited one or more learning disabilities. Other students were placed in the program by the parents to improve skills. Each child who participated in the practicum will be described in detail further on in this chapter.

The staff at the setting included teachers, all of whom were Florida certified with prior teaching experience. These teachers also were graduate students who had specific responsibilities and duties and who operated in one of the following capacities: general

classroom teacher, language arts teacher, computer science teacher, or administrative intern. Specifically, the teacher breakdown was as follows: thirteen general classroom, three computer science, and three administrative interns.

The author of this practicum served as a computer science teacher and wrote a daily log called the Student Skill Development Log (Appendix A:50) for each student. The log tracks the author's daily objectives and responsibilities in the summer program. The author maintained daily communication with the pupils' general teacher; and, if applicable, the language arts instructor. Daily informal meetings with the students' teacher(s) took place before or after school, as needed. Formal group meetings were held weekly. These meetings allowed for a broad exchange of ideas and/or concerns. The weekly meetings also served as an inservice to promote the infusion of critical thinking skills into the various curriculum areas.

The author brought several specific teaching experiences to the program. The author had taught for a total of seven years in Grades 1 through 12. Most of this time was spent with middle and high school age

children. The author was state certified in general science, physical science, chemistry, and biology. Over the course of the author's teaching career, such topics as curriculum relevance, advisement intervention, interdisciplinary studies, and career awareness were professionally investigated by the author through inservice and graduate-level college courses. The writer, accustomed to working with low level students through previous teaching assignments, affirmed the concept that students with or without learning difficulties deserve the opportunity for the highest degree of personal achievement. Such a learning opportunity is available to the students in this summer school where the atmosphere and professional attitude were highly positive towards and supportive of children.

In general, class size during the summer session was kept to five pupils per general instructor, allowing for increased personal attention for each student during the three-hour school day. The general classes also utilized precision teaching, which gives students constant feedback on their progress while allowing pupils to work at their own pace and to

achieve at their individual success level. The computer science classes had a student-teacher ratio of 2:1, providing even greater personalized student attention.

The computer classes were all held in one room simultaneously. The lab equipment included: six Apple IIc computers and monitors and five Panasonic printers. Each student worked on a separate computer most of the time. Three teachers each interacted with student groups of two with the following grade separations: Grades K through 1, 1 through 3, and 3 through 6. The grade separations were determined by the needs of the program as dictated by the number of students in each grade level. A generally equal distribution of students to computer science teachers resulted from the grade grouping, providing each teacher with 14 to 16 students. The author personally instructed 16 students. The arrangement of two students per teacher allowed each pupil to have individualized computer instruction. The educational atmosphere of the computer lab and the general classroom was established with the concept of providing the student a calm, yet visually motivating opportunity to achieve.

Common plans and decisions benefiting all children, regardless of grade level, in the computer science program were discussed and determined through the joint decisions and actions of all three instructors. The cooperation among the teachers in the program resulted in the following:

1. Maximum space was provided for each student, while discouraging interference between groups through a space oriented hardware arrangement.

2. The software library at the school site was doubled for the represented grade levels.

3. The bulletin boards and mobiles were designed to reinforce computer literacy, attract student interest, and promote desired behavior and attendance.

4. A unified reward plan was implemented to encourage student effort. The plan was fair and attainable for all students who participated in the computer classes regardless of grade level or ability.

Although all students were impacted by their involvement in the computer program, six students, in particular, were selected to participate in the target group for this practicum. The children were selected partially on their availability to attend computer

classes daily. Each student was scheduled for 20 minutes of computer class per day with the exception of the children taking verbal language skills. On the days that a child missed 20 minutes of general class time for verbal language development, the student was not permitted to miss any additional class time to attend computer class. The nine students who were not involved in the verbal skill development were chosen to be tested for the target group. The six students who had the strongest conception of number values between 0 and 10 were chosen to be the target group for this practicum. The evaluation tool used is found in Appendix B:88. A brief description of each student follows:

Student 1, age eight, a male entering the second grade, made friends easily and had a positive attitude. This student demonstrated grade level math reasoning in general class but had trouble with basic subtraction beyond the number nine. The student had not been identified as learning disabled.

Student 2, age eight, a female entering the second grade demonstrated first-grade skills in overall reading comprehension. The student related well with

peers. This student's parents felt that the student needed more self-confidence in decision making. The student's math level skills were lower than the reading comprehension skills, and the student had difficulty transferring and applying a concept to a new situation. Math application was very low even though a learning disability was not recognized.

Student 3, age nine, a female entering third grade, had a diagnosed learning attention deficiency and was on medication. The student's IQ was in the normal range. The student's slight loss of hearing and attention span problems may have been linked to the stroke that was sustained when she had heart surgery as an infant. This child also wore glasses because of some visual impairment.

Student 4, age nine, male entering third grade, was retained in kindergarten because of age and overall immaturity. The child's IQ was in the normal range, and two previous tests for learning disability were negative. The student was visually impaired and wore glasses. The student worked well in the lab.

Student 5, age eight, was outgoing and self-motivated. The pupil was physically impaired from

cerebral palsy and was placed in an ESE program during the regular school year. This student found subtraction difficult and relied heavily on counting on his fingers.

Student 6, age eight, used fingers to solve grade level math problems. This child had a poor self-concept and projected a pessimistic attitude towards personal success. The pupil was familiar with addition and subtraction of numbers 1 through 18.

The children in the target group were judged by their general class teachers as being below grade level in their ability to compare numbers. The author set out to clarify the competence level for each student in this skill through a series of pretests (Appendix C:90). The students as a whole were found to be below the accepted base level of competency. The minimal accepted level of competence, as determined by general testing of accomplished grade level students, is 30 completed problems out of 60 with 80 percent correct in one minute. The State of Florida supported the local county's curriculum. This curriculum stated that second-grade students had to have a grasp of the numerical values and symbols of the numbers 1 to 50.

Third-grade students were expected to be capable of recognizing and comparing the values of 1 to 100. The grade level textbooks titled Riverside Mathematics written by Haenisch, Hill, and Merril (1987) further supported these skill levels in their objectives related to the skill of comparing numbers.

The writer's opinion is that the skill increases will be accomplished through the use of effective student-teacher-computer interactions. The following objectives will be accomplished during the six-week computer-based implementation of the practicum experience:

1. After four weeks of computer instruction, all of the six students in the target group will be able to identify and match the greater than (>), less than (<), and equal to (=) signs with their corresponding meanings at least 90 percent of the time as measured by a paper and pencil test (Appendix D:95).

2. After six weeks of computer instruction, all six of the students' ability to compare the value of two numbers from 0 to 100 as being less than, greater than, or equal to given number(s) will improve as

determined by an increase of at least 20 percent on the posttest as compared to the pretest.

3. After a three-week period of computer instruction, the students in the target group will improve their ability to compare the values of open mathematical sentences. The mastery of this skill will be determined by comparing the pretest and posttest scores with the goal being a 20 percent or more increase in the percentage of correct answers.

4. The six students will, after using computers for six weeks, individually rank computer-assisted learning with other teaching methods as to student preference on a pen and paper survey (Appendix E:97).

CHAPTER II

Research and Solution Strategy

Researchers agree that numerically-based problem solving can be taught to children (Costa, 1985; Channel, 1984; Matsumoto, as cited in Costa, 1985) and must be initiated on a continuing basis in the elementary years (Barnard, 1986). Problems that children are likely to encounter first in the primary grades are numerical in nature. Progrow, as cited by Costa (1985), stated that after basic numerical operations have been combined as sentences and mastered, words are added to the problems to provide realism in mathematical applications to the real world. This process begins in about Grade 4 with basic operation mastery expected by the end of Grade 7.

The review of the literature did not disclose an exact, comprehensive process through which problem solving with numerical comparisons and addition or subtraction operations (NCAS) can be effectively taught to children. Parnes, as cited by Costa (1985:230), explained why there is no exact process:

> It appears that some people have experiences that develop their facility in intellectual processes associated with creativity and intelligence. Research seems to demonstrate that we can design educational programs for many of those experiences, rather than merely waiting and hoping for them to happen.

Children do not always come to school with previous outside experiences that will enhance their ability to learn mathematical problem-solving concepts, such as NCAS. Educators have to design programs that are flexible in filling the gaps of intellectual experience absent in some students. Parnes, as cited in Costa (1985), further classified these gaps in a child's experience in terms of the following five perceptual blocks:

1. A difficulty exists in finding the problem or task.

2. There is an inability in defining the problem.

3. The child lacks the skill of narrowing the intellectual task to solve the problem.

4. The student's observational skills are limited in using input from the five senses.

5. The individual is affected by cultural and emotional factors which interfere with perception and the thinking process.

Overcoming the perceptual blocks must be one of the objectives of a comprehensive mathematical program.

The computer as an educational tool can help overcome these blocks by meeting individual student needs (Bitter, 1981; Matsumoto, as cited in Costa, 1985). A decade ago, the National Council for Teachers of Mathematics realized the potential of the computer to permit the development of new strategies for assisting students in learning numerically related problem-solving skills (Channell, 1984). No other technology (Martin, 1981) has greater potential for responding to an individual's personal learning patterns and unique learning style.

Students are able to overcome some of their cognitive blocks because they are captivated by computers for the reasons (Vickers, 1984) that follow:

1. The machines are patient.

2. The student can continually be given positive reinforcement.

3. Computer programs can have complete recall and are able to show the student where previous errors were made.

4. The machine does not get bored and neglect the student.

5. Students are fascinated by a machine that responds to their thoughts in a positive manner.

The computer (Progrow, cited in Costa, 1985) is effective in developing students' cognitive skills by providing instant and continuous feedback to the learner. Teachers are generally not able to provide such support due to class size and curriculum constraints. Computers acting as the student's personal instructor (Wiederhold, as cited in Costa, 1985) can sequentially present lesson material to students and follow up on student errors.

Computers will capture the attention of students, however, they cannot produce cognitive gain without quality teaching or structured curricular strategies (Progrow, as cited in Costa, 1985). A program designed to develop a student's skills in numerical problem solving cannot consist of just the computer and its software. According to Progrow, the teacher operates

as a facilitator who gives students the mental space and the responsibility to think about and solve problems. The traditional teaching strategies do not provide the student such an opportunity. The role of the teacher in computer-assisted instruction (CAI) is to oversee the students as to the use of the computer and its feedback. The students can make decisions about the quality of their strategies in solving particular problems. The process, in addition to building fundamental skills using higher cognitive levels (Progrow, as cited in Costa, 1985), is a powerful method to make students feel good about themselves because of the continuous positive response to the student's individual intellectual growth.

As important as the teachers and machines are in the computer-assisted instructional environment, the software is the heart of the program. Properly chosen software will stimulate ". . . thinking and self-reflection in ways that encourage students to progress from processing ideas in serial fashion to simultaneous or parallel thinking" (Matsumoto, as cited in Costa, 1985:249). With appropriate software, students will extend and apply mathematical and similar abstract

concepts to real world problems. The computer with the appropriate software becomes a fascinating ". . . mirror for the user's thought processes, enabling him to reflect on his own thinking and improve his metacognitive skills" (Seeley-Brown, as cited in Costa, 1985:252).

According to Brevil (1987), the software chosen for the machine can motivate the learning disabled student if the following criteria are adhered to:

1. The program format must be non-threatening thus boosting the user's self-esteem while competing against a student's feeling of failure.

2. Immediate feedback and reinforcement must be given to reassure the student and raise the student's confidence level.

3. Drill and practice applications must be made interesting through graphic and game formats.

4. Several learning modes of perception must be appealed to so that students can interact with the software through their strongest mode instead of their weakest mode of perception.

Beyond software that entertains and applies specific skills while promoting student interaction,

computer programs are now available that diagnose cognitive flaws in solving specific problems. Bernard and Sandberg (1986) reported that a group of regular elementary school children used a CAI program that considered cognitive processes in early mathematics. The results were that one-third of the group increased significantly in skill level. The program identified strategies that children developed to solve open sentence mathematical problems. The program also helped instructors identify where an individual's thinking was incorrect. CAI programs further individualize a student's education by increasing the student's and the teacher's awareness of the student's cognitive weaknesses.

Common in regular classrooms today are students who possess a degree of Attention Deficit Disorder (ADD). Finding software that will benefit these students is difficult. As Terwilliger (1986:3) described them, ". . . they are of at least normal intelligence and can be very personable and inquisitive in a one-to-one learning situation, sharing every child's delight in exploration and discovery." In a group, however, they are demanding of attention, seldom

on task, and unable to focus attention. Terwilliger (1986) also stated that ADD children often miss problem cues and lack the strategies that are learned and taken for granted by children in general. Appropriate software for ADD students must cater to both their social and cognitive needs in order to expand their problem-solving capacity.

While attempting to meet students' special needs, mathematical applications in software have developed in complexity over the last 15 years (Haase et al., 1985) but have not mastered the instruction of multi-step problems. There is no debate that the computer's large memory and analytical power can, with the correct software, create a flexible learning environment that will respond to the needs of the individual student. According to Haase (1985), this software will eventually (a) anticipate student misconceptions, (b) detect error patterns through student responses, and (c) keep a permanent record of student progress.

With the volume of software either available or entering the educational market today, educators want to know how to select the best software for a given situation. One way of selecting software is to rely on

someone else's guidelines. There is a service program (Cradler, as cited in Costa, 1985) called Project CompuTHINK that can assist educators in evaluating software and related cognitive skills. In establishing one's own evaluation process, computer science journals, magazines, and software publishers can provide assistance in developing specific guidelines.

Consider that it may not be necessary or desirable to be very specific in the purchase of software, particularly if the software is modifiable. Eiser (1986:27) when addressing the problem of finding software for special education students stated:

> Happily, more and more publishers are producing programs with flexibility built into the basic structure - flexibility that may include increasingly difficult levels of challenge, puzzles whose complexity can be modified, adjustable feedback mechanisms and more.

The writer of this practicum is in complete agreement with the statements of the authors whose views were discussed in this chapter. In developing the method to implement the practicum, their thoughts were considered.

The author will follow Eiser's (1986) recommendation of utilizing modifiable software that is adaptable to the needs of the individual child. In adapting and choosing the software, the whole child's learning experience, as recommended by Parnes (cited in Costa, 1985), will be considered. Whenever possible, software will be chosen that helps the student narrow the intellectual task, increase observational skills, and overcome negative emotional perceptions to decrease problem-solving difficulty.

The software chosen and classroom atmosphere constructed will incorporate Brevil's (1987) and Vickers' (1984) recommendations in being (a) non-threatening to the student, (b) immediate in positive feedback, (c) interesting and rewarding of creative thinking, and (d) exciting to students by providing several perceptual modes of response--visual, auditory, and tactile.

The method for developing the CAI program will assume that each child has some degree of learning attention deficit. The author will seek to exemplify Progrow's concept of the teacher as facilitator. The author will oversee each student on a one-to-one basis

as needed, thus providing the student with mental space and allowing the young person the responsibility for individual thought. The freedom of a student to think and make errors without perceived harmful consequences is indicated by the research as being superior to teacher-directed, time-oriented, and curriculum-centered instruction.

CHAPTER III

Method

The practicum began before the summer with a meeting of all of the staff and administrators involved in the summer program for students with learning problems. At that initial orientation meeting, the author met other teachers, many of whom were also involved with their own practicum research. A bond formed among the teachers. Each was conducting individual research, and this helped foster a cooperative attitude towards and respect for one another's instructional and research needs over the course of the summer program. The program would run from 9:00 a.m. to 12:00 p.m. Monday through Friday.

At the orientation session, the principal, who was later designated as the author's mentor, reviewed the goals and structure of the program. The staff was made aware that the students as a whole exhibited various learning problems including but not limited to lack of attention span and poor self-esteem. The 10 general classroom teachers who assisted in language and

mathematical skill development were directed to coordinate their efforts with the three computer teachers. Class lists that were distributed to the general teachers and the computer teachers would later integrate the 52 students from Grades 1 through 6 into the program according to guidelines later agreed upon. At the completion of the orientation, a general contract specifying computer staff member responsibilities (Appendix F:99) was signed by the author.

The author observed an evening class of representative students even though the observation was optional for computer teachers. The experience was helpful to the author because it demonstrated a student centered program where the atmosphere was positive and friendly. The students were kept from getting bored by the continual changing of activities every five to 10 minutes. The students were often rewarded with praise and tangible rewards, while the teachers exhibited patience and eagerness in their interaction with the students. The timed and non-timed activities performed by the students related primarily to the visual and tactile perceptual learning modes, although

story-telling and several learning machines appealed to the auditory mode. Throughout the observation session, the author became convinced that the software utilized in the program needed to have more than one perceptual mode for student-machine interaction.

The computer science teachers were excused from the next two orientation sessions concerning precision teaching and next met together in late June. Two days later, the author examined the designated computer science classroom to assess the classroom atmosphere, hardware, and software needs.

The brightly lit room with soothing off-white walls was of standard size and designed for up to 20 students. There were two wall-size blackboards and one bulletin board. In addition to a teacher desk, two student-size tables with four accompanying chairs stood near the window side of the room. Along two walls were wall-size shelves with a total of seven computers, three printers, and sufficient plug sockets. One of the computers was not compatible with the others and needed the 3 1/2 inch disk format as opposed to the 5 1/4 inch format. Four stools and six student chairs were also in the room. The author felt that the room

was adequate for the program but needed a few additions and modifications to help motivate students and stimulate critical thinking.

The author, along with the other computer science teachers, purchased bulletin board materials and mobile cut-outs. The bulletin boards included class charts that were used to track and reward attendance on Fridays. Students who came to all scheduled classes and behaved properly were permitted to interact with computer programs of their choice. Another bulletin board demonstrated the different types of keys on the computer keyboard along with a computer's major parts and their functions. The mobiles and the bulletin boards, caught the eye and described how computers worked, communicated, and were used. Stamps and stickers were also purchased by the author to reward students' efforts and behavior. By the first day of the session, the computer lab room was decorated, the schedule organized, the software evaluated, and class rules established.

The students were scheduled according to grade level by being split into three groups. On the average, 17 students were assigned to each computer

teacher. The author was personally responsible for 12 boys and four girls in the first, second and third grades. The author developed a student schedule that consisted of eight groups of two students, each receiving a 20-minute block of computer time. Similar schedules were adopted by the other computer science teachers. The schedules allowed each student to work on a separate machine for a maximum and equal amount of time. Other parameters that had to be considered in student scheduling were the day camp starting time, the days of language skill class, and the students' grade level. Students who were identified as day campers were scheduled to attend computer class prior to 11:00 a.m. Students involved with language skill classes were scheduled for computer class on Monday, Wednesday, and Friday. The computer instructors agreed to follow these scheduling guidelines to simplify the overall school scheduling for the program. The author intentionally scheduled action research subjects of identical regular class teachers and program requirements for the same time slots so that evaluations given to the students would go unnoticed by the remaining students.

The computer science teachers agreed to list and support five class rules within a comprehensive discipline plan that students from all groups were to follow. The rules as posted included:

1. Treating one another properly is a must.
2. Eating and drinking in the lab are not permitted.
3. Fooling around is not permitted.
4. Yelling or screaming are not permitted.
5. Seating must be on time.

Rewards for adhering to the rules were: computer time for alternative game programs, behavior stickers, positive messages to take home, and candy. The students enjoyed computer time to work on their favorite programs and often asked for behavior stickers. The author experienced no major behavioral problems. The negative consequences included: less computer time for part of or the whole computer period, phone calls home, parent conferences, and referrals to the administration. The selective use of a negative consequence would depend both on the nature and the repetitiveness of the negative behavior.

On the first day, the author escorted the students to the computer science classroom from their regular classroom. The author would continue to escort the students as needed during the summer program if they were more than one minute late to class. The first day of class served as an orientation period to introduce the students to the computer science program, review class rules, and discuss teacher expectations. The author began by introducing all the computer science teachers. The students were told where their work areas would be and were shown how to boot a disk. The pupils were informed as to the kinds of game-type software that they would use. The author's students were given the following initial weekly schedule for software activities:

Monday - mathematical concepts.
Tuesday - language skills.
Wednesday - mathematical concepts.
Thursday - language skills.
Friday - open lab, student choice depending on behavior

The author viewed the initial changing of software subject areas as motivational because it provided

variety. The shortness of computer lab periods made the use of multiple subjects difficult within the limitation of 20-minute periods.

The open lab period on Friday served several important functions. The period provided an opportunity for students to be creative and to develop language skills while promoting parental involvement. The students often used a program that allowed them to interchange picture and words anywhere on the screen. Upon completion, a hardcopy was printed to be posted or taken home. Parent involvement was achieved by encouraging students to bring their parents to visit the computer center and by providing students with hard copies of creative work (Appendix G:102) to take home. Students were also occasionally given a supportive activity to do at home with a parent. A bulletin board to display student work was constructed just outside of the classroom to encourage students to bring their parents to the room. As a result, parents occasionally stopped in to see the computer center.

The first 1 1/2 weeks were used to assess students' individual skill levels in mathematics, content reading, and grammar. The assessment period

allowed the author to determine the proper software for each student and the degree of emphasis to place on particular areas of instruction. Although the practicum implementation dealt with mathematical concepts, the instructor was also responsible for using appropriate software to support the general teacher's efforts to build language skills.

The author kept a daily log (Appendix A:50) for each student in the practicum group. The logs were used to review student's progress with parents, general class teachers, and administrators. The logs also served as one source of information for making decisions about using, changing, or modifying of the software. The students' apparent emotional states were recorded and given serious consideration. Boredom, frustration, and failure were dealt with as soon as they were encountered. The students' degree of success in software "games" and the teacher's evaluations of the students were recorded by the author as raw scores. Software effectiveness and raw scores were reviewed by the author at the end of each day.

The daily log specified software that the target group used: Arithmetic Critters (Minnesota Educational

Computing Corporation (MECC), 1986), Counting Critters (MECC, 1985), Early Addition (MECC, 1984), Speedway Math (MECC, 1986), Number Munchees (MECC, 1984), Minus Mission (Developmental Learning Materials, 1986), and Path Tactics (MECC, 1985). The programs as a group were highly modifiable, allowing the author to address specific skills with individual students. Skill areas covered by the computer programs included: number recognition of numbers 0 to 100; and addition, subtraction, and inequalities of numbers 0 to 20. By manipulating the program parameters, the degree of difficulty was adjusted after each "game." Sound effects and speed were also manipulated. Thus, the software remained challenging to the student while continuing to allow achievement.

Flashcards were made by the author and used during the first week of class to familiarize students with the inequality and equality symbols. Each group of two students took part in a symbol drill during the first five minutes of four class sessions. All curriculum related student interaction, aside from the use of flashcards, involved software or evaluations.

Three major types of evaluations noted in the daily log were utilized to derive raw scores. All were paper and pencil tests. The first test, which consisted of four evaluative instruments on single- and double-digit numerical comparison of varying difficulty (Appendices B:88 and H:105), were administered during the second, fourth, and final weeks of the practicum implementation. The evaluation during the fourth week was used to determine the effectiveness of assigned software. School site permission for the use of individual evaluative instruments was previously granted by the Orange County Public School Precision Teaching Project of Orlando, Florida. The author timed the paper and pencil evaluations for one minute, using a stopwatch. The author sought but could not locate an available instrument to evaluate the students' ability to compare the values of simple open mathematical sentences, therefore, an evaluation instrument was developed by the author (Appendix I:109). This second evaluation was given during the fourth and final weeks of the practicum implementation with a three-minute time period for completion of the test. The data received from the two aforementioned test types were

used to determine any needed changes to the implementation plan including: student grouping, activity scheduling, and software choosing. The author's conclusions, which were based on the acquired data, were also shared with each student's general class teacher.

The data that was collected justified software modification and title changes in several instances when the skills of addition or subtraction of double-digit numbers 10 through 20 were not mastered. Student grouping was unaffected because the students worked effectively together in a spirit of friendly competition, often communicating with each other in challenging situations by exchanging helpful hints. Ongoing observations of student progress and evaluations led the author to modify the schedule of software activities during the final three weeks of implementation. The modification resulted in two additional days of computer augmentation of math skills and the elimination of language skill reinforcement on Tuesdays and Thursdays. The open lab on Friday continued so the author could reward student effort.

The third type of evaluative instrument required the students to identify the symbols for greater than, less than, and equal to (Appendix D:95). This author prepared the questions on this evaluation, and the test was given during the last week of implementation. Three minutes were allowed for this multiple-choice test because the students could not write quickly.

The author made an effort to eliminate test environment problems and test anxiety. All necessary materials were provided, and the writer chatted briefly with the students in a friendly manner before the evaluation. Students were encouraged to do their best at all times. The tests were never assigned a letter or number grade beyond that of a raw score. Test results were not interpreted beyond the raw score for comparison of skill development. The author believed that not ranking student progress helped minimize test taking anxiety. The students were informed that rating or grading of individual students was not the intent of the tests. In reviewing the results with students, the author emphasized their accomplishments and not their difficulties. Several students continued to exhibit nervousness in taking the tests even though they were

told that the purpose of the evaluations was to assist the instructor in helping individual students. In order to preserve the objectivity of individual students, the pupils selected for the target group were never informed of their selection.

A survey of student instructional preference (Appendix J:111) was developed by the author and was given during the final week of the course to determine how students felt about computer-assisted instruction as compared to other instructional methods. The survey was read to the students, and they were given 10 minutes to complete it. Students ranked each of five instructional methods on a scale of one to five with one being most liked and five being least liked.

At the end of the program, the author, along with the other computer science teachers, assisted in the field day exercises and presented all of the students with a commercially available award for completion of the computer science summer program. The author also made verbal recommendations of specific software to the parents of each child in the hope that this would help the child's continued academic development. These recommendations were based on the author's notes in the

daily student log and were made in the presence of the general classroom teacher at the final student progress evaluation meeting. A formal write-up for the parents on the student's academic progress in the computer program was provided to several parents who had home computers or desired pupils to continue in the fall program. The summary was provided on the software recommendation sheet (Appendix K:113) supplied by the administration.

CHAPTER IV

Results

In this chapter, results obtained from this practicum are compared to the stated objectives. Intervening variables affecting outcomes will also be discussed. The first objective established a minimum level of competence of 90 percent in identifying the following mathematical symbols of comparison: less than (<), greater than (>), and equal to (=). As can be seen in Table 1, two students did not achieve the stated percentage of accuracy. The results do show that four, or two-thirds of the students, did satisfy the objective. Thus, the first objective for the group was not achieved. Several factors contributed to the unacceptable scores of the two students.

After the evaluation, Student 6 told the author that confusion occurred regarding the English words for the symbols. The student had been absent for two days immediately prior to the testing, which may have caused the student's problem with the symbols and their

written meanings. Student 3, with the diagnosed attention deficiency, may have exhibited a lapse in attention during the exam because test items missed were answered correctly in other areas of the exam.

Table 1

Identifying Less Than, Greater Than, and Equal To Symbols - Posttest Scores

Student	Posttest Scores Answers Correct	Percent
1	10	100%
2	10	100%
3	7	70%
4	10	100%
5	10	100%
6	2	20%

The ability of students to compare any two numbers between 0 and 100 did increase for each student by more than 50 percent demonstrating that the students had a stronger grasp of number values and the use of comparative symbols. Table 2 below reflects combined total scores of a four-part pretest and posttest. The data indicates that the second objective of this practicum was met. Even students who initially had a

weak grasp of number values showed considerable growth during the implementation period. The table does not reflect that the smallest increase in the understanding of numerical quantities involved numbers between 50 and 100. The weakness in this area may have been because the author's modification to the software did not place enough consistent stress on numbers greater than 50. The students preferred numbers less that 50 because the numbers were less frustrating to work with at the slowest game pace. As a result, the author spent less than four class sessions on numbers greater than 50. Had the implementation period been longer, the students' ability to compare two numbers might have increased.

Table 2

Number Comparison (0-100)

Student	Pretest # Correct	Posttest # Correct	Percentage Change
1	79	124	57%
2	66	107	62%
3	24	88	266%
4	100	157	57%
5	35	112	220%
6	45	74	64%

The third objective required that all of the students increase their ability to compare open mathematical sentences by at least 20 percent. Table 3, on the following page, shows that only one student met the goal of increasing by 20 percent the ability of comparing open mathematical sentences involving one plus or minus mathematical operation on each side of an equality. The main reason for the lack of success in meeting this objective may have been the student's reliance on memory as opposed to making effective notations on scratch paper when carrying numerical values. A second reason for not meeting the objective was that the author's attempts to instruct the students about how to efficiently estimate differences in value did not transfer to test situations. The two aforementioned skill deficiencies in combination served to adversely affect the results of open sentence comparison. This caused students to attempt less problems with more errors during the time provided. Student 2 and Student 6 continued to have difficulty with double-digit subtraction even though computer time devoted to the skill areas was increased.

Table 3

Comparing Open Sentences of Addition and Subtraction

Student	Pretest	Posttest	Percent Change
1	18	20	+11%
2	25	8	-17%
3	18	20	+11%
4	11	17	+55%
5	22	14	- 8%
6	12	12	0%

The comparison of Tables 2 and 3 demonstrates that computer-assisted instruction of single-step mathematical problem solving involving critical thinking is accomplished more readily than multiple-step problem solving with critical thinking. This trend reinforces the author's view that considerable more student-computer contact time than was allotted was needed to reinforce such a critical thinking activity.

Objective four required students to evaluate the various instructional methods that were used with them in the general, language, and computer classes. The students were asked to apply higher order thinking skills to rank instructional strategies according to personal comfort, enjoyment, and perceived, positive

outcomes. The results of the Student Instructional Methods Survey shown in Table 4 show that students generally preferred computer-assisted learning to other instructional forms. Lower numbers indicate the students' greater preference of an instructional method. The use of the computer was, on the average, preferred by all the students above the other educational methods. Learning by writing on paper was judged by students to be the next best method. Students least preferred to learn by using a stepboard.

Table 4

Instructional Methods: Student Preferences

Instructional Method	Average Student Preference (*)
1. learn by working with a story in a book	3.83
2. learn by working on a blackboard	3.16
3. learn by working on a sheet of paper	2.33
4. learn by working with computer software	1.50
5. learn by working with a stepboard	4.16

* The lower the combined average preference score is, the more desired the particular instructional activity is to the average student.

The results, although not conclusive, demonstrate that student-computer interaction has a place in the

development of critical thinking skills as applied to mathematical concepts. All of the students in the target group increased their ability to identify symbols and compare two numbers with values between 0 and 100. The fact that one student significantly improved the ability to solve and compare open mathematical sentences incorporating addition or subtraction of whole numbers was not conclusive because of limitations built in to the research design; for example, the small size of the target group and the limited class period time. The author further feels that student preference for computer interaction over other instructional forms will be supported in the future by further research.

CHAPTER V

Recommendations

Additional research is needed to help educators to effectively establish computer-assisted math programs. More research about computer-assisted learning in the major disciplines is needed in the elementary, middle, and high school settings for average, below average, and gifted achievers. The author would like to witness research on the effect of computer-assisted learning on other basic as well as advanced skills of the subject areas: mathematics, language arts, and reading. Research also needs to be done on the effectiveness of computer-assisted learning on the mastery of scientific applications where many skills are interacting simultaneously.

This author would suggest two changes in the future design of this practicum. First, the target group should be increased to between 12 and 20 students. This would provide a larger body of subjects from which to extrapolate data. The basis for success could then be based on 80 percent of the students

demonstrating a 20 percent or more improvement in a given skill area. Examination of the results would then allow for extraneous variables that were not accounted for in this study. Second, the contact time with the students should be increased to between 30 and 40 minutes per session. Larger time blocks would enhance lesson continuity by permitting introduction, self-discovery, practice, skill transfer, and evaluation, or summarization comfortably within one session.

The information gathered from research and the results of the action research implementation are being shared on a local level with administration and staff members at the school where the author is employed. The practicum experience has been discussed with administration in the context of purchasing computer systems for the various subject areas of the school. Additionally, practicum results are being communicated both formally and informally to faculty members through department and team meetings, faculty lounge discussions, and inservice classes for computer literacy and software infusion. The author believes that teachers with a computer science education

background must assume leadership roles within their districts and individual institutions to facilitate the change from passive group education to individual active education now possible with computer software.

The writer will seek ways to present both the results and applicable research from this practicum at district-wide and regional meetings. Also, the writer will continue to serve as an advocate for the use of computer-assisted instruction to enhance basic skills and critical thinking within the context of existing elementary and secondary level programs.

Reference List

Barnard, Y. and Sandberg. J. "Development of a CAI Program Within the Field of Early Mathematics." 1986, ERIC ED 273 481.

Bitter, Gary. "Microcomputers and Problem Solving." 1981. ERIC ED 218 705.

Brevil, Margarette. "Using Microcomputers to Help Learning Disabled Students With Arithmetic Difficulties." 1987. ERIC ED 289 697.

Channel, Dwayne E. "Problem Solving With Computers." Mathematics Teachers, October 1984, pp. 534-541.

Costa, Arthur L. (Ed.) Developing Minds Resource Book. Alexandria, VA: Association for Supervision and Curriculum Development, 1985.

Developmental Learning Materials. Minus Mission. (Computer Program) Allen, TX: Author, 1986.

Eiser, Leslie. "'Regular' Software for Special Kids?" Classroom Computer Learning, October 1986, pp. 26-35.

Haase, H. Marion R. and Mestre, J. "Design Features of Pedagogically-Sound Software in Mathematics." 1986. ERIC ED 290 627.

Haenisch, Siegfried, et al. Riverside Mathematics. IL: Riverside, 1987.

Jones, Daisy M. Curriculum Targets in the Elementary School. New Jersey: Prentice-Hall, 1977.

Martin, Kathleen. "The Learning Machines." Arithmetic Teacher, November 1981, pp. 41-43.

Minnesota Educational Computing Corporation. Arithmetic Critters. (Computer Program) St. Paul, MN: Author, 1986.

Minnesota Educational Computing Corporation. Counting Critters. (Computer Program) St. Paul, MN: Author, 1985.

Minnesota Educational Computing Corporation. Early Addition. (Computer Program) St. Paul, MN: Author, 1984.

Minnesota Educational Computing Corporation. Number Munchees. (Computer Program) St. Paul, MN: Author, 1984.

Minnesota Educational Computing Corporation. Path Tactics. (Computer Program) St. Paul, MN: Author, 1985.

Minnesota Educational Computing Corporation. Speedway Math. (Computer Program) St. Paul, MN: Author, 1986.

"See-Mark Tests," Precision Teaching Project, Orange County Public Schools, Orlando, FL, n. d.

"See-Write Tests," Precision Teaching Project, Orange County Public Schools, Orlando, FL, n. d.

Terwilliger, Cynthia. "The Child With Attention Deficit Disorder: Finding a Safe Place to Learn Problem Solving." 1986. ERIC ED 290 272.

Vickers, Melanie. "Reset Your Button and Drive into Computer-Assisted Math." Academic Therapy, March 1984, pp. 465-471.

Weiss, Aaron. Math Rabbit. (Computer Program) Menlo Park, CA: The Learning Co., 1986.

Appendices

Appendix A

Student Skill Development Logs

Appendix A

Mr. Schuman
Teacher

STUDENT: _____

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

Appendix A

STUDENT: STUDENT #1 2+ 9:00-9:20AM Mr. Schuman Teacher

DATE:
SKILL: -

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/26
SKILL:

ORIENTATION

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/27
SKILL: + -

SOFTWARE: Arithmetic critters
OBSERVATIONS: + - (0-18) PTS 13

RESULTS: + D software not challenging enough

.....

DATE: 6/28
SKILL: + -

SOFTWARE: speedway math
OBSERVATIONS: levels 1+2 all correct.

RESULTS: did well on 10 group problems (at 5 min)

.....

DATE: 7/29
SKILL: CREATIVITY

SOFTWARE: NUMBER MUNCHERS
OBSERVATIONS: picked + used [illegible] creating challenges (seemed interested in what another student was doing).

RESULTS:

BEST COPY AVAILABLE

STUDENT: STUDENT #1

Mr. Schuman
Teacher

DATE: 7/2/90
SKILL: - subtraction -

SOFTWARE: Speedway Math: Big Race
OBSERVATIONS: takes time to fill in single subtraction problems, scanned
scored 96/10 in 10min/53 SEC. name placed in Hall of Fame
RESULTS: in disk.
move to 2 place dig. + +

DATE: 7/2/90
SKILL: Reading MICRO GRAPHS GRADE 3 TOP READERS CLUB. WORLD SOUND RECOGNIT. "WORD MUNCHERS" #153

SOFTWARE:
OBSERVATIONS: - answered 3 TYPO rapidly + correctly. scored less than 70% on level
RESULTS: Comp'leted (continue) -

DATE: 7/4/90
SKILL:

SOFTWARE:
OBSERVATIONS: Holiday

RESULTS:

DATE: 7/5/90
SKILL:

SOFTWARE: WORD MUNCHERS
OBSERVATIONS: - worked more quickly, battle strategy, less mistakes
enhanced coordination, of vowel sounds indicated within
RESULTS: Score 180 gained 75 pts. typed words.

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

STUDENT: STUDENT #1

Mr. Schuman
Teacher
N-26 26/0
N-28 25/0
N-30 12/0
N-32 9/9

Correct Responses

DATE: 7/9/90
SKILL:
SOFTWARE: >,< - Problem Solving
OBSERVATIONS: 4, 60 second tests
RESULTS: 79/79

.....

DATE: 7/10/90
SKILL: Building words (blends, diagraphs)
SOFTWARE: MECC 179 Shares time
OBSERVATIONS: captivated by program
RESULTS: completed 9 words.

.....

DATE: 7/11/90
SKILL: 1) Building #'s out of groups of 10 and ones - worksheet 1-1.
2) less than with +, -
SOFTWARE: number Munchers level 2
OBSERVATIONS: quickly grasped building #'s out of groups of tens and ones.
RESULTS: scored 180 on less than with +, - problems.

.....

DATE: 7/12/90
SKILL:
SOFTWARE:
OBSERVATIONS: Absent
RESULTS:

.....

DATE: 7/13/90
SKILL: creativity: picture + story
SOFTWARE: KIDPIX
OBSERVATIONS: - manipulated picture elements
RESULTS: painted @ little assistance

Appendix A

STUDENT: STUDENT #1

Mr. Schuman
Teacher

DATE: 7/16/90
SKILL: Inequalities 1-20, (+)
SOFTWARE: Large Numbers
OBSERVATIONS: attentive
RESULTS: highest Score = 125

Reviewed Flash Cards < , > Comparing single + double digit = s.

.....

DATE: 7/17/90
SKILL: Inequalities with +, - (10-20)
SOFTWARE: Number Munchers
OBSERVATIONS: - adapted quickly
- performed operations quickly.
RESULTS: High Score: 125

late to class 10 minutes.

.....

DATE: 7/18/90
SKILL: < >
SOFTWARE: Math Rabbit 0-99 high speed Program 2
OBSERVATIONS: - rapid consistent decisions
= high self confidence in this task.
RESULTS: 90+% accuracy

.....

DATE: 7/ /90
SKILL: < >
SOFTWARE: Math Rabbit 0-99 high speed Program 2 Review
OBSERVATIONS: = student was confident and quick in applying skill
= student handled 2 digit numbers with only 4 errors
RESULTS: best/worst/ errors in 55 problems.

Mid test
Comp/Approx
M-6 3 0
M-28 5 0
M-30 3 1 2
M-32 7 1 0

Mid test = increased speed + perform

.....

DATE: 7/20/90
SKILL: GAME / CRITICAL THINKING = STRATEGY
SOFTWARE: MINUS MISSION
OBSERVATIONS: - enjoyed program
- worked out strategy
RESULTS: 6 Error

Critical Thinking Exercise

For Indexes.

Appendix A

STUDENT: STUDENT #1

Mr. Schuman
Teacher

DATE: 7/23/90
SKILL: + − ≠ 10-20
SOFTWARE: Number Munchers
OBSERVATIONS:

RESULTS:
SCORE 190

/ Pretest + 18/1 (+, −, ≠, >, <, =)

.....

DATE: 7/24/90
SKILL: + − ≠ (0-20) < > =
SOFTWARE: # Munchers
OBSERVATIONS: = played 3 complete games
= Practiced well especially working out easy problems.
= unsure of adding/- double digit #'s.
RESULTS:
High Score 130

.....

DATE: 7/25/90
SKILL: + − <, >, =, ≠ (10-20)
SOFTWARE: # Munchers & Inequalities
OBSERVATIONS: - worked properly + efficiently
- used effective board strategy
- answered accurately
RESULTS:
High Score 335

.....

DATE: 7/26/90
SKILL: + − <, >, =, ≠ (10-20)
SOFTWARE: # MUNCHERS MECC #170 / MATH ASSIST for >, <
OBSERVATIONS: - worked consistently < Enables him simple errors while working under time pressure.
High Score 275
RESULTS:
→ ready for post test.

.....

DATE: 7/27/90 ONE CLASS
SKILL: CREATIVE DRAWING
SOFTWARE: RAINBOW PAINTER
OBSERVATIONS:
= brought up picture elements
= deleted picture elements
= moved elements around screen.
RESULTS:
Printed.

< = > + −
MID TEST
Computation Problem solving
Score 20/2 (+, −, <, >, =)

STUDENT: STUDENT #1

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: SUBTRACTION, STRATEGY

SOFTWARE: PATH TACTICS (SUBTRACTION)
OBSERVATIONS: -quickly grasped strategy
-made no subtraction errors
RESULTS:
↳ enjoyed program

post-test comparison
N-26 40/0
N-28 32/0
N-30 25/0
N-32 27/0

late to class

DATE: 7/31
SKILL: SUBTRACTION STRATEGY

SOFTWARE: Path Tactics
OBSERVATIONS: -only had time to set up program
RESULTS:
was disappointed that we did not have time to play ↳ 20/0

determine Determination
~~Sign Identification Evaluation~~
Addition/SUBTRACTION/INEQUALITY
COMPOUND ARCADE POSTTS

DATE: 8/1
SKILL: Add Subtract / Game Strategy
Student-Student Interaction
SOFTWARE: Path Masters
OBSERVATIONS: interacted in a friendly helpful competitive manner
RESULTS: made no math errors, used game strategy
↳ enjoyed game, won on 14 moves

MALES

DATE: 8/2
SKILL:

Absent

SOFTWARE:
OBSERVATIONS:

RESULTS:

DATE: 8/3
SKILL:

Field Trip

SOFTWARE:
OBSERVATIONS:

RESULTS:

Appendix A

STUDENT: STUDENT #2 2+ 4:00-9:20 Mr. Schuman Teacher

DATE:
SKILL: -

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/26
SKILL:

SOFTWARE:
OBSERVATIONS: ORIENTATION

RESULTS:

.....

DATE: 6/27
SKILL: + -

SOFTWARE: counting critters
OBSERVATIONS: + -(0-17) PTS 13

RESULTS: recognizing #'s, simple adding
RT

.....

DATE: 6/28
SKILL: counting

SOFTWARE: counting critters
OBSERVATIONS: confused counting pts. per critter + Early Addition to unit 1 student wrong once gave feedback at end of problem type (1 + 5 = 7)

RESULTS:
Continue with next level of Early Addition

.....

DATE: 6/29
SKILL:

SOFTWARE:
OBSERVATIONS: late (9:16 AM)

RESULTS:

Appendix A

STUDENT: STUDENT #2

Mr. Schuman
Teacher

DATE: 7/2/90
SKILL: - Addition -

SOFTWARE: Early Addition
OBSERVATIONS: 50% of words (+) correct, cannot add. 3 numbers sum.

RESULTS: continue to enroll 3 Early Addition.

.....

DATE: 7/3/90
SKILL: SPELLING

SOFTWARE: SPELLING WISARETTE
OBSERVATIONS: word list #1 - spelled many wrong letters, wrong keys, 1/2 time with words longer than 5 letters.

RESULTS: program problem - stuck on 1st word and is lost. Program is not interactive enough.

.....

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS: Holiday

RESULTS:

.....

DATE: 7/5/90
SKILL:

SOFTWARE: Top Leaders Clubs.
OBSERVATIONS: - read 17 words

RESULTS: = out of 17 questions, 14 min read correct/3 min

.....

DATE: 7/6/90
SKILL: Subject Area Testing

SOFTWARE:
OBSERVATIONS:

RESULTS:

Appendix A

STUDENT: STUDENT #2

Mr. Schuman
Teacher

DATE: 7/9/90
SKILL: >< - Problem Solving, Pretest
SOFTWARE:
OBSERVATIONS: 4, 60 sec tests
RESULTS: 66/68

N=16 12/16
N=23 21/23
N=10 16/16
N=32 26/32

.....

DATE: 7/10/90
SKILL: reading comprehension
SOFTWARE: MECC #173 Those Amazing Reading Machines
OBSERVATIONS: - identified are inadequate
- got bored reading, needs more verbal reinforcement.
RESULTS: - switched around in program. continue 6 middle level-3

.....

DATE: 7/11/90
SKILL: 1) Building #s out of groups of 10 and ones. - worksheet N=15
2) less than with #'s
SOFTWARE: Number Munchers level 2,
OBSERVATIONS: at first, was unsure of self, in doing worksheet, gained confidence
RESULTS: software problem. number Munchers - would not recognize
Scores: 3rd Qtr.: Num. with ... Step ... process (she began to understand the reasoning and ... (not ... problems ... 5 and 2 ... 5 & 2).

DATE: 7/12/90
SKILL: blends + digraphs
SOFTWARE: MECC #179 Phonics Power Time - Blends + Digraphs.
OBSERVATIONS: - student felt it was easy
RESULTS: 13 correct / 2 wrong 86.6% on level 2.

.....

DATE: 7/13/90
SKILL: creative: picture + story
SOFTWARE: Bald Headed Chicken
OBSERVATIONS: - manipulated, several different elements, needed help constructing sentence description
RESULTS: printout @ much assistance

Appendix A

STUDENT: STUDENT #2 — Mr. Schuman, Teacher

DATE: 7/3/90
SKILL: Inequalities 1-20 (<, >)
SOFTWARE: Number Munchers
OBSERVATIONS:

RESULTS: Highest Score 75

.....

DATE: 7/7/90
SKILL: Inequalities @ +, −, (#'s 1-20)
SOFTWARE: Number Munchers
OBSERVATIONS: - slow at basic operations
- confused in process of solving problem

RESULTS: High Score: 20

.....

DATE: 7/8/90
SKILL: < >
SOFTWARE: Math Rabbit, 20-75, medium speed (Program 2, Level 2)
OBSERVATIONS: - made many errors at first (50% correct)
- finished last two sequences with 90% correct.

RESULTS: increased # correct by using a solution strategy (comparing first #'s

.....

DATE: 7/9/90
SKILL: < >
SOFTWARE: MATH RABBIT 0-99, Medium Speed (Program 2, Level 2)
OBSERVATIONS: = errors produce more errors → lack of confidence
= confusion on 2 digit # comparisons

RESULTS: = Inconsistent Reasoning

.....

DATE: 7/20/90
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

Appendix A

STUDENT: STUDENT #2

Mr. Schuman
Teacher

DATE: 7/23/90
SKILL:

Absent

SOFTWARE: -
OBSERVATIONS:

RESULTS:

.....

DATE: 7/24/90
SKILL: + - (10 20) < > = ≠
SOFTWARE: # MUNCHERS Inequalities, MECC #170
OBSERVATIONS: - used paper + pencil
RESULTS: - proceeded slowly + carefully.
scored 300 or less 3 repeated games.

pre-test
25/25
finished 2:57

.....

DATE: 7/25/90
SKILL: + - < > = ≠ (10-20)
SOFTWARE: # Munchers Inequalities MECC #170
OBSERVATIONS: - works slowly
- confuses - and + signs in some meanings
RESULTS: 14 ghost Score 575
- gets confused under time pressure

5 leave room for 1 more with 5 minutes left

.....

DATE: 7/26/90
SKILL: + - < > ≠ = (10-20)
SOFTWARE: # MUNCHERS MECC #170
OBSERVATIONS: - confusing + and -
- not assimilating > < ≠
RESULTS:
High Score 80

MATH-MAGIC < > = ≠ (20-50)
- did well misses only 1 or 2
- used 3 perfect games.
70% accurate.

.....

DATE: 7/27/90
SKILL: creative Drawing Object calc
SOFTWARE: Rainbow Valley
OBSERVATIONS: - manipulated picture
RESULTS: - put elements of picture
together
- enjoyed program.

MAD + - cs + conjoined inside < = > + -
- DNE 25/25/3 min.

Appendix A

STUDENT: STUDENT #2

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: Adding, subtracting (board game)

SOFTWARE: Path Tactics
OBSERVATIONS: - does not understand strategy - not planning moves.
- adds well, makes mistakes subtracting.

RESULTS:
computer won twice

.....

DATE: 7/31
SKILL: subtracting

SOFTWARE: Path Tactics
OBSERVATIONS: - better use of spatial board strategy
- subtraction errors

RESULTS:
computer won -

Instructional Preference - Session
Sign Identification Evaluation Basic Addition/Subtraction/Inequality Posttest - Compound Problems SCORE 8/

.....

DATE: 8/1
SKILL: Add/subtract/game STRATEGY
NEGATIVE INTEGRATION

SOFTWARE: Path Tactics
OBSERVATIONS: - made errors with random #'s, used inflative board strategy
- was not cycling @

RESULTS: - behavior indicated she had a good time
- last game, demonstrated proper competitive conduct.

.....

DATE: 8/2
SKILL: Off-Abs

SOFTWARE: Path Tactics
OBSERVATIONS: played against partner student

RESULTS:
enjoyed experience worked well together

.....

DATE: 8/3
SKILL: Field Trip

SOFTWARE:
OBSERVATIONS:

RESULTS:

BEST COPY AVAILABLE

Appendix A

STUDENT: STUDENT #3 9:40-10:00AM Mr. Schuman Teacher

DATE:
SKILL:

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/26/90
SKILL:

SOFTWARE: ORIENTATION
OBSERVATIONS:

RESULTS:

.....

DATE: 6/27
SKILL:

SOFTWARE: COUNTING CRITTERS Menu #: 1.95
OBSERVATIONS: + - (0-17) ITS 2

RESULTS: +A (said she liked it)

.....

DATE: 6/29/90
SKILL: 4, >

SOFTWARE: Math Rabbit, Menu 2, Level 2, speed 1
OBSERVATIONS: difficulty recognizing > @ #'s greater than 20

RESULTS: 50% accuracy, continue next week.

.....

DATE: 6/29/90
SKILL: creating pictures

SOFTWARE: LIMA Items
OBSERVATIONS: learning to move objects

RESULTS: 1 picture made.

Appendix A

STUDENT: STUDENT #3

Mr. Schuman
Teacher

DATE: 7/2/90
SKILL:

SOFTWARE: Math Rabbit, Stairwell down game
OBSERVATIONS: with help of ascending order of tens chart

RESULTS: 75/100, some in determining < or > with numbers 20 → 100

.....

DATE: 7/3/90
SKILL: Matching vowel sounds

SOFTWARE: WORD MUNCHER
OBSERVATIONS: student found + differentiated first to match vowel sounds to words

RESULTS: improved 100% 20 → 40. (continues)

.....

DATE: 7/4/90
SKILL:

SOFTWARE:
OBSERVATIONS: Holiday

RESULTS:

.....

DATE: 7/6/90
SKILL:

SOFTWARE: WORD MUNCHERS
OBSERVATIONS: - less frustrated - enjoyed making (full of Fine (with teacher help)

RESULTS: 26, 40, 35

student does not use strategy to work ... determine ... sound out words ... away ...

.....

DATE: 7/6/90
SKILL: Subject Area Testing

SOFTWARE:
OBSERVATIONS:

RESULTS:

Appendix A

STUDENT: STUDENT #3 — Mr. Schuman, Teacher

DATE: 7/9/90
SKILL:
SOFTWARE:
OBSERVATIONS: Pretest 7L, 4 60 M minute tests
- student has trouble drawing > or < signs
RESULTS: N=26 6/10, 4/6 N=28, 6/15 N=30, 7/8 N=32

DATE: 7/10/90
SKILL: making words from sounds
SOFTWARE: MECC #179 blends + digraphs
OBSERVATIONS: - sounded out words
- prefers cooking over making milkshakes.
RESULTS: 4 words 3/6, 4/5, 6/3, 2/7 (continue)

DATE: 7/11/90
SKILL: 1) Building #'s from tens & ones
2) Finding #'s 1-20
SOFTWARE: Number Munchers level 2
OBSERVATIONS: 1) student does not comprehend how tens and ones
RESULTS: making a #
2) student did well on limited # problem, + only
HW. complete
N-15 by Fri.
go up to level 3

DATE: 7/12/90
SKILL: making words from sounds
SOFTWARE: MECC #179 Blends + Digraphs
OBSERVATIONS: = does not sound out words.
= ranked level 2 on secondary
RESULTS: 5/9

DATE: 7/13/90
SKILL: word I.D. activities

SOFTWARE:
OBSERVATIONS: Parent words
= completed activities
RESULTS: - identified words on word list.
Abstracted from text.

check for N-15

Appendix A

STUDENT: STUDENT #3 — Mr. Schuman, Teacher

DATE: 7/12/90
SKILL: <, > 1-20, (+)
SOFTWARE: Number Munchers' [illegible] 1-20 (+)
OBSERVATIONS: - [illegible] / Reviewed
- played game repetitively
RESULTS: Flash cards <, >
↳ 220 high score — Number comparisons, sizes [illegible]

DATE: 7/ 7/90
SKILL: <, >, 10-20 (+, -)
SOFTWARE: Number Munchers
OBSERVATIONS: - student has difficulty reading numbers of 2 or more digits and relating to values
- student is weak in +, - of 2 digit #s.
RESULTS:
High score: only 20. / reviewed number comparisons, Flash cards <, > 10-20

DATE: 7/18/90
SKILL: <, >
SOFTWARE: Math Rabbit (Program 2)
OBSERVATIONS: - frustrated with number comparisons over 10
- is not consistant in solving the equality problem
RESULTS:
50% accuracy

continue

DATE: 7/19/90 ← Review sheets 10 min. then computer time
SKILL: <, >
SOFTWARE: MATH RABBIT (PROG) 0-99, slow
OBSERVATIONS: - performed consistently poor +
RESULTS: frustrated, interrupts on [illegible] numbers
= many errors 9/10 problems. all [illegible]

still having difficulty with numbers

DATE: 7/20/90
SKILL: —, # guessing
SOFTWARE: SPACE SUBTRACTION
OBSERVATIONS: - had difficulty w/ 2 digit #'s
- could perform operations with organized assistance
RESULTS:
↳ mixed
↳ she enjoyed game

Appendix A

STUDENT: STUDENT #3

Mr. Schuman
Teacher

DATE: 7/23/90
SKILL: multiple problem solving

SOFTWARE: #Muncher MECC #170
OBSERVATIONS: - guessed
- no strategy
- not self motivated
RESULTS: - enjoys the pictures

late START = excused (missed picture taking).

PRETEST SCORE: 4/24 = 17%

Home work: A+ Addition sheet.

* Student cannot add single digit #'s

DATE: 7/24/90
SKILL:

Absent - brainstorming activity

SOFTWARE:
OBSERVATIONS:

RESULTS:

DATE: 7/25/90
SKILL: +, - single digit problems.

SOFTWARE: MECC #787 Early Addition
OBSERVATIONS: - mixes addition & subtraction
- often uses counting strategy
- skill = single digit
- inconsistent use of visual clues

RESULTS: Mixed 60% correct 12 of 20

MECC #165 counting Cattle
Go Logo Format two

MECC #166 Arithmetic single
Go Logo format two single

DATE: 7/26/90
SKILL: < > = (10-20)

SOFTWARE: MATH RABBIT < > a# 20-50
OBSERVATIONS: did well getting most all of them correct.

RESULTS: - she work quickly + accurately on this one-step comparison.

#MUNCHER MECC #170 (10-20)
- has trouble performing multiple tasks, she forgets what to do first.
- trouble subtraction single digit #'s.

DATE: 7/27/90 OPEN LAB
SKILL:

SOFTWARE: WORD MUNCHERS
OBSERVATIONS: - difficulty in guessing word played guessing
RESULTS: - frustrated and encouraged to continued motivation.

MIDTEST
COMPLEX PROBLEM SOLVING SCORE =
< > - - =

8/7/3 Min

Appendix A

STUDENT: STUDENT #3

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: ~~Subtraction~~ gave strategy to ⇒ addition/strategy
SOFTWARE: Path Master
OBSERVATIONS: - student has great difficulty subtracting
- student can add single one digit numbers up to 12
RESULTS: continuation sum of 12, then counts on fingers but was frustrated @ lack of general progress

compare post test
N-30
N-40
N-526

DATE: 7/31
SKILL: Addition/strategy
SOFTWARE: Caterpillar 5+
OBSERVATIONS: less hesitant
- counted better
RESULTS: → played game well

Instructional Preference determination
Sign Identification Evaluation
Score 7/10
(needs to do combined skill posttest

DATE: 8/1
SKILL: Addition/strategy/Peer Interaction
SOFTWARE: Path Masters
OBSERVATIONS: - performed add operations on single digits correctly
- more aware of guessing strategies
RESULTS:
- lost game but had a good time

Add/sub/Inequality
Combined skills
posttest.
20/17/3

DATE: 8/2
SKILL: of SW - AB: choice
SOFTWARE: Do you this!
OBSERVATIONS: - tried spelling new vocabulary words @ Step and Redo
- needed help @ regularly
RESULTS: → was interested + concentrated @ Program

DATE: 8/3
SKILL:
SOFTWARE:
OBSERVATIONS:
RESULTS:

Field Trip

Appendix A

STUDENT: STUDENT #4 6/27 Mr. Schuman
Teacher

DATE:
SKILL: -
SOFTWARE:
OBSERVATIONS:
RESULTS:

.....

DATE: 6/26
SKILL: ORIENTATION
SOFTWARE:
OBSERVATIONS:
RESULTS:

.....

DATE: 6/27
SKILL: + -
SOFTWARE: #7 EARLY ADDITION
OBSERVATIONS: + - (0-18) 1'TS 5
RESULTS: 20/20 correct on 1, 19/20 on #3
.....

DATE: 6/28/90
SKILL:
SOFTWARE: Early Addition
OBSERVATIONS: on screen #5, 90% accuracy, 3 level single digit + addition
RESULTS: move to subtraction, next time, and two digit addition.

.....

DATE: 6/29/90
SKILL: creative word association
SOFTWARE: Bean Story Picture Development
OBSERVATIONS: used graphic elements with descriptive words
RESULTS: wanted to repeat experience.

STUDENT: STUDENT #4

Mr. Schuman
Teacher

DATE: 7/2/90
SKILL: -

SOFTWARE: SPEEDWAY MATH
OBSERVATIONS: subtraction - regrouping problems 96 correct took > 15 min.
RESULTS: accurate but unsure - uses fingers to count to complete
try to wean off fingers & increase speed.

DATE: 7/2/90
SKILL: Main Ideas (Identifying)

SOFTWARE: Reader's Treasure Chest Series 2
OBSERVATIONS: -3 right 3 wrong - needed assistance reading / lacks confidence / student
RESULTS: student could not teach to concentrate. writes up words to finders

DATE: 7/4/90
SKILL:

SOFTWARE:
OBSERVATIONS: Holiday
RESULTS:

DATE: 7/5/90
SKILL: Sentence structure, determining meaning

SOFTWARE: write, Rabbit
OBSERVATIONS: - student reads more practice interpreting meaning in sentences
RESULTS: - needs to picked out who, what, where, when, why

DATE: 7/6/90
SKILL:

SOFTWARE:
OBSERVATIONS: ABSENT
RESULTS:

Appendix A

STUDENT: STUDENT #4 — Mr. Schuman, Teacher

DATE: 7/9/90
SKILL: pretest ≥ < # comparisons
SOFTWARE:
OBSERVATIONS: N-26 N-28 N-30 N-32
RESULTS: 22/0, 28/0, 30/0, 20/1

.....

DATE: 7/10/90
SKILL: reading comprehension
SOFTWARE: MECC #1, 88 Mavis Beacon Reading Machine IV
OBSERVATIONS: at first he was not reading machine descriptions for meaning and missed 2 out of 2 then reading the instructions carefully and
RESULTS: better understanding the program he got 2 correct

.....

DATE: 7/11/90 (continues)
SKILL: 1) building # s out of tens and ones - N-15 problems sheet.
2) practicing keyboard, - problems. - software
SOFTWARE: MECC Number Munchers
OBSERVATIONS: 1) did very well on building #s from tens and ones, 100;
RESULTS: 2) did correct 2) did poorly on game program: strategy was 50, 25, 25 the main reason for low score. not in complete

.....

DATE: 7/12/90
SKILL: reading comprehension
SOFTWARE: MECC #188 Reading Machine STV
OBSERVATIONS: - lacks patience in reading descriptions
- is not assimilating information
RESULTS: out 2 stores correct

.....

DATE: 7/13/90
SKILL: - percent subtraction
SOFTWARE: #1 Numbers
OBSERVATIONS: - late to class 15 minutes
RESULTS: scored 90 on 0-10 (#1's)

Appendix A

STUDENT: STUDENT #4

Mr. Schuman
Teacher

DATE: 7/16/90
SKILL: inequalities 1-10 (+)
SOFTWARE: # Munchers Meeting 90
OBSERVATIONS: - played competitively
RESULTS: - [illegible] score 295

Reviewed Flash cards: < > symbols single + double digit number Compare

DATE: 7/17/90
SKILL: <, >, #10-20 (+, -)
SOFTWARE: Munchy Numbers
OBSERVATIONS: - did not adapt to game changes in a timely manner
- was frustrated @ lack of progress
RESULTS: - misread comparison at top.
High Score: 30

Reviewed <, > w/ # pairs did 100%

DATE: 7/18/90
SKILL: < >
SOFTWARE: Math Rabbit Level 2
OBSERVATIONS: - completed all tasks through 99 and left game
RESULTS: - placed in 2nd Math Rabbit Level 2 (gave for added challenge)

DATE: 7/19/90
SKILL: < >
SOFTWARE: Math Rapid Level 2 Prog 2, 0-99, Test +
OBSERVATIONS: - worked effectively, moved at acceptable pace
- completed with little assistance
RESULTS: only 2 errors on 32 problems.

Min to do # comp
N-16 59%
N-13 47%
N-0 69%
N-12 35/2

10% of time

# of Ps + results: maintaining speed and accuracy

DATE: 7/20/90
SKILL: — missing #s.
SOFTWARE: SPACE SUBTRACTION
OBSERVATIONS: - engaged on task
RESULTS: - completed mission

BEST COPY AVAILABLE

Appendix A

STUDENT: STUDENT #4 — Mr. Schuman, Teacher

DATE: 7/03/90
SKILL: and logic problem solving
SOFTWARE: #4 Numbers Inequality, +-, 10 to 20.
OBSERVATIONS: - complained about easy problems
RESULTS: - lazy in working on problems, motivation to give proper attention
- requires additional motivation to give proper attention
Score 10.0

DATE: 7/24/90
SKILL:
SOFTWARE:
OBSERVATIONS:
RESULTS:

Absent: Brainstorming Activity

DATE: 7/25/90
SKILL: +, −, <, >, =, ≠ (10 to 20)
SOFTWARE: # NUMBERS INEQUALITY MECC #170
OBSERVATIONS: - used improved board strategy
- figured out each problem individually
RESULTS: HIGH SCORE 235

DATE: 7/26/90
SKILL: < > = ≠ (10-20) + −
SOFTWARE: MATH LAB 3 B/T
OBSERVATIONS: - made several careless errors
RESULTS: - worked quickly

NUMBERS MECC #170
= attributed the game to learning strategy → maintained concentration & positive attitude to
HIGH SCORE 200

DATE: 7/27/90
SKILL: CREATIVE PICTURES
SOFTWARE: RAINBOW VALLEY
OBSERVATIONS: - manipulated pictures
RESULTS: - very creative in designing castle
→ enjoyed result

Post Test
COMBINED PROBLEM SOLVING
SCORE
23 - 7
3 MIN

Appendix A

STUDENT: STUDENT #4

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: Addition/Subtraction GAME STRATEGY
SOFTWARE: PATH MASTER
OBSERVATIONS: = planned
=
RESULTS: computer won 2 games

11th conference (1 MIN)
first test
N-26 40/0
N-27 42/0
N-30 38/4
N-32 37/3

DATE: 7/31
SKILL: Addition, Game Strategy
SOFTWARE: Path Master
OBSERVATIONS: - set up quickly
- figured out game strategy
RESULTS: beat computer once

Instructional Reference
Sign Identification Evaluation
10/0

DATE: 8/1
SKILL: Add / Game strategy w/ Integers
SOFTWARE: Path Master
OBSERVATIONS: - fed/read out pace with multiple choices
- was at first reluctant to interact
RESULTS: won game.

DO TOMORROW
Adding, subtracting
Expanded...

Add/Subtract/Integers
Comprehension
Post test
7/31

DATE: 8/2
SKILL: Open Labs = choice
SOFTWARE: Oregon Trail
OBSERVATIONS: - engaged peer program
RESULTS: - made choices
- worked well Kimberly

DATE: 8/3
SKILL:
SOFTWARE:
OBSERVATIONS: Field Trip
RESULTS:

Appendix A

STUDENT: STUDENT #5 6/27/90 Mr. Schuman
Teacher

DATE:
SKILL: -

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/26/90
SKILL:

SOFTWARE: ORIENTATION
OBSERVATIONS:

RESULTS:

.....

DATE: 6/27/90
SKILL: (+ -) Addition

SOFTWARE: Early Addition # 788
OBSERVATIONS: Name 1 adding #'s 100's PTS 7
Name 2

RESULTS:

.....

DATE: 6/28/90
SKILL: + Addition

SOFTWARE: Early Addition # 789 Name 3, 4
OBSERVATIONS: Name 3 - 100%
RESULTS: Name 4 - 100% takes long time to add - uses fingers.
Began Name #5 - 100%

.....

DATE: 6/29
SKILL: Creative, coordination.

SOFTWARE: BEAR STORY, PICTURE
OBSERVATIONS: developed picture from elements
finished picture

RESULTS:
enjoyed

Appendix A

STUDENT: STUDENT #5 — Mr. Schuman, Teacher

DATE: 7/2/90
SKILL:
SOFTWARE: Early Addition Module 1+5
OBSERVATIONS: Adding 3 #s., 88% correct
RESULTS: slow., completed Throughroom 5 → could not move to next room because of slow coordination

DATE: 7/3/90
SKILL: Reading Comprehension
SOFTWARE: MICROGRAPHS. READING CLUBS GRADE (4)
OBSERVATIONS: - student gets flustered, reads stories too fast, but can handle the grade level.
RESULTS: achieved 2nd floor first building — continue

DATE: 7/4/90
SKILL:
SOFTWARE: Holiday
OBSERVATIONS:
RESULTS:

DATE: 7/5/90
SKILL:
SOFTWARE: Reading Clubs Grade 4
OBSERVATIONS: - enjoyed program overcame frustration of getting some questions wrong.
RESULTS: - got to 4th floor of 1st building - level 1 - too hard — continue — Read Clubs Grade 3

DATE: 7/6/90
SKILL: Subject Area Testing
SOFTWARE:
OBSERVATIONS:
RESULTS:

Appendix A

STUDENT: STUDENT #5 — Mr. Schuman, Teacher

DATE: 7/9/90
SKILL: -> L problem solving & comparisons
SOFTWARE:
OBSERVATIONS: Pretest
RESULTS: 8/5, 6/4, 13/22, 8/18
N=20 N=28 N=30 N=32

.....

DATE: 7/10/90
SKILL: Reading Comprehension; matching machines to text
SOFTWARE: Mac Utility Disk Descriptions.
OBSERVATIONS: - worked up to level 4, got bored at end of part 4
- requires more reinforcement and feedback.
RESULTS: did well on matching machines 11/12 3/0

.....

DATE: 7/11/90
SKILL: 1) building numbers (with, +, -)
2) not equal to
SOFTWARE: MECC Number Murder
OBSERVATIONS: 1) confused on building numbers by groups of ten or so; missed all ten
2) finished with program, corrected her errors & complete 16 for man
RESULTS: scored 45, 35, 115 program of not equal to.

.....

DATE: 7/12/90
SKILL: Reading Comprehension
SOFTWARE: MECC #178 Ready Machines
OBSERVATIONS: 1) student given paper + pencil to enhance reading comprehension 3) disappearing helpful sentences
2) student more sure of self
4) working faster.
RESULTS: finished 4 machines in 16 minutes

Check homework

.....

DATE: 7/13/90
SKILL:
SOFTWARE:
OBSERVATIONS: Absent
RESULTS:

Appendix A

STUDENT: STUDENT #5

Mr. Schuman
Teacher

DATE: 7/12/90
SKILL: L, 7, #1-20 (+)
SOFTWARE: # Munchers, MECC #170
OBSERVATIONS: - student made errors based on counting thinking
RESULTS: highest score 285

/ review - flashcards
L, 7 synthesis
comparing sizes

DATE: 7/17/90
SKILL: L, > #10-70 (+, -)
SOFTWARE: # MUNCHERS, MECC #170
OBSERVATIONS: - uses fingers first in performing how operations proper
- not totally positive using pattern (hand-eye coordination)
- occasionally counts wrong
RESULTS: level 1 High Score 90 → makes comparisons well.

DATE: 7/23/90
SKILL: L 7, 0-99
SOFTWARE: Math Rabbit Reg 2 (#0-99) Reteach
OBSERVATIONS: - student quickly guessed choices.
- student got bored and was given Number Munchers (negatively reinforced)
RESULTS: Student ready for mid test.

DATE: 7/19/90
SKILL:
SOFTWARE: Math Rabbit Reg 2 (#0-99) Re-test
OBSERVATIONS: - student made errors, 50% of time
- student was frustrated
- student cannot subtract after they have made.
RESULTS: → student needs more practice

Mid test - (M
worked faster
No change in skill

DATE: 7/20/90
SKILL: #60 BEGINNING
SOFTWARE: SPACE SUBTRACTION
OBSERVATIONS: - entered program
RESULTS: → did well on program. double digit subtraction.
Assign H.W.

ONE on 1

Appendix A

STUDENT: STUDENT #5

Mr. Schuman
Teacher

DATE: 7/23/90
SKILL: expand problems + - negatives 10-20
SOFTWARE: # Munchers MECC #170
OBSERVATIONS: - worked very slowly
- made frequent errors
RESULTS: - was attentive
scored 30.

Retest Score
22/27
3 Min

DATE: 7/24/90
SKILL: +, -, <, >, =, 10-10 expand problems
SOFTWARE: # Munchers MECC #170
OBSERVATIONS: - work applications of problems on paper, tried to take shc.
- figured out scores of problems by subtracting.
RESULTS:
Highest Score 60 (reviewed solution strategy @ start)

DATE: 7/25/90
SKILL: +, -, <, >, =, ≠, (10-20)
SOFTWARE: #MUNCHER INEQUALITIES
OBSERVATIONS: - was not figuring out problems before copying answer or using of how to the equation shown at the top.
- not thinking through answers.
RESULTS:
scored 15, 25, still low

DATE: 7/26/90
SKILL: + - < > ≠ = (10-20)
SOFTWARE: MATH RABBIT (CO-OP)
OBSERVATIONS: - made several errors at first and then became more proficient.
RESULTS:
understood comparisons

MUNCHERS MECC #170 (10-2
- successfully accessed.
- worked better when using her fingers
SCORED 110

DATE: 7/27/90
SKILL: CAPTIVE
SOFTWARE: RAINBOW VALLEY
OBSERVATIONS: - made changes elsewhere
- manipulated elements
RESULTS:
- printed
enjoyed process.

MATH Computer Activities
Scoring
4/5

SCORE:
3 Min.

STUDENT: STUDENT #5

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: Addition, Game Strategy
SOFTWARE: Path Masters
OBSERVATIONS: - lack of strategy; 2 addition errors
- errors in judgement
RESULTS: lost 2 games, got frustrated

.....

DATE: 7/31
SKILL: Addition Subtraction Strategy
SOFTWARE: Path Masters
OBSERVATIONS: - used effective strategy
- no addition errors or subtraction errors
RESULTS: - lost one game - got motivated to play more
- won one game.

.....

DATE: 8/1
SKILL: Addition / strategy
SOFTWARE: Cat Master
OBSERVATIONS: - wanted to get even @ computer
- [illegible] to solve multiple step problems
RESULTS: won game & teacher assistance.

.....

DATE: 8/2
SKILL:
SOFTWARE: Oregon Trail (Hi Res) Program
OBSERVATIONS: - used to decision making
- able to read at program level.
RESULTS: - did not finish program

.....

DATE: 8/3
SKILL:
SOFTWARE:
OBSERVATIONS:
RESULTS:

Field Trip

Appendix A

STUDENT: STUDENT #6 6/27/90 Mr. Schuman
Teacher

DATE:
SKILL: - -

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/26
SKILL: ORIENTATION

SOFTWARE:
OBSERVATIONS:

RESULTS:

.....

DATE: 6/27
SKILL: + -

SOFTWARE: [illegible] COUNTING [illegible] FARS #165
OBSERVATIONS: # recognition + counting [illegible]

RESULTS: Menu 1 100%
2 "

.....

DATE: 6/28/90
SKILL: + -

SOFTWARE: Early Addition #738
OBSERVATIONS: Menue 3 - 100%
Menue 4 - 100%

RESULTS:

.....

DATE: 6/29
SKILL: Mapping creature.

SOFTWARE: MAZE
OBSERVATIONS: management: score 3400 on level 1.

RESULTS: slow reader, enjoyed program

STUDENT: STUDENT #6

Mr. Schuman
Teacher

DATE: 7/2/90
SKILL:

SOFTWARE: GREGORIAN MATH
OBSERVATIONS: added errors, not fast enough to do gave up → switched to game
RESULTS: has trouble with ordering operations + #'s

DATE: 7/2/90
SKILL: ~~MICROGRAPHS~~ reading comprehension
SOFTWARE: MICROGRAPHS READING CLUB GRADE 3
OBSERVATIONS: some student frustration over reading + answering ques. - inaccurately.
RESULTS: student on level A got to 3rd floor. Continue

DATE: 7/3/90
SKILL:

SOFTWARE:
OBSERVATIONS: Holiday

RESULTS:

DATE: 7/4/90
SKILL:

SOFTWARE:
OBSERVATIONS: absent

RESULTS:

DATE: 7/5/90
SKILL: absent

SOFTWARE:
OBSERVATIONS:

RESULTS:

BEST COPY AVAILABLE

Appendix A

STUDENT: STUDENT #6 — Mr. Schuman, Teacher

DATE: 7/9/90
SKILL: problem solving.
SOFTWARE:
OBSERVATIONS:
RESULTS: N=10 9/0, N=9 10/1, N=10 12/1, N=32 14/0 = comparison / exist

DATE: 7/10/90
SKILL: Reading comprehension - relating machines to description.
SOFTWARE: MECC #107 Reading Madness IV
OBSERVATIONS: student found it difficult at first to picture what he was reading.
RESULTS: 0/2, 1/2 working at level 2 — student enjoys reading

DATE: 7/11/90
SKILL: 1) building #s out using numbers worksheet N=12
SOFTWARE: MECC # Numbers
OBSERVATIONS: 1) did well on worksheet N=12, while several errors
RESULTS: 2) scored 80 on inequality program with ranges of 1-20. enjoyed program.

DATE: 7/12/90
SKILL: Reading Comprehension
SOFTWARE: MECC # 108 Reading Madness IV
OBSERVATIONS: - student given paper + pencil to draw out what he reads
- student reading with more concentration.
RESULTS: identified 2 machines. (used diagram generated by student to help identify)

DATE: 7/13/90
SKILL: making a picture which describes a story (creative) 2/15
SOFTWARE: Counting Chickens
OBSERVATIONS: - creative, discipline, using elements of fiction
RESULTS: completed, after a critiqued program

Appendix A

STUDENT: STUDENT #6 — Mr. Schuman, Teacher

DATE: 7/16/90
SKILL: L, > #1-20 (+) / Review: flash cards; L, > symbols; comparing single — double
SOFTWARE: #, Numbers & NECC#170
OBSERVATIONS: - did not program the accuracy before starting the test.
RESULTS: - made errors in selecting comparisons. Highest Score: 220
.....on ~~accuracy~~ quickly ~~frustrated~~.....

DATE: 7/17/90
SKILL: L, > (#10-20), + −
SOFTWARE: Number Munchers.
OBSERVATIONS: - works +, − @ double digits accurately
- solves problems orderly (first the bot operation, then the comparison)
RESULTS: Level 2: HIGH SCORE: 220.
.....

DATE: 7/18/90
SKILL: <, >
SOFTWARE: Math Rabbit (0-99) Program 2 Level 2
OBSERVATIONS: - student within 50% to 100% correct over 15 minute period
RESULTS: - Student ready for post test on capacities
to continue @ Math Rabbit, Number Munch- L, >, +, −, <
.....

DATE: 7/19/90
SKILL: <
SOFTWARE: Math Rabbit (0-99) Prog 2 Level 2 fastest +
OBSERVATIONS: - student worked efficiently and accurately
RESULTS: - student ~~got bogged~~ ~~page~~ ~~Program~~ ~~worksheets~~ increased speed on speed
Math test 1/10 N-16 1/11 N-20
1/10 N-28 1/10 N-32
.....

DATE: 7/20/90
SKILL: Logic Gates
SOFTWARE: Mind Benders
OBSERVATIONS: - organized attack
- attempted a more challenging puzzle.
RESULTS:

↳ learning experience
he scored very well.

Appendix A

STUDENT: STUDENT #6

Mr. Schuman
Teacher

DATE: 7/23/90
SKILL: compound problem solving +, −, compound inequalities (10-20)
SOFTWARE: # Munchers MECC #170
OBSERVATIONS:
RESULTS: = worked relatively independently
= made careless errors (not thinking logically in my opinion)
- moved slowly.

1.2) 2) 1
Score
12/1

DATE: 7/24/90
SKILL: compound problem solving +, −, <, ≤, ≥, (10-20)
SOFTWARE: # MUNCHER INEQUALITIES
OBSERVATIONS: - wanted to quit program - did something else
- student influenced in solving problems
- student grades assignments as warranted by problems.
RESULTS: High Score 90

teacher plan: attempt using in terms of real

DATE: 7/25/90
SKILL: +, −, <, >, =, ≠ (10-20)
SOFTWARE: # MUNCHER INEQUALITIES
OBSERVATIONS: = better grasp of problems, knowing strategies
= figured out problems before working computations.
- enjoying program more because he had better control over it
- played 3 games
RESULTS: HIGH SCORE 190

DATE: 7/26/90
SKILL: +, −, <, >, ≤, ≥, = (10-20)
SOFTWARE: MATH ROBBIT
OBSERVATIONS: - moved
- made several careless errors
- concentrated and did precise problem computation when wrong
RESULTS:

MUNCHER, MECC #170
- student did better using his effective strategies < knowledge of his goals
- stuck to self good +
HIGH SCORE 165

DATE: 7/27/90
SKILL:
SOFTWARE:
OBSERVATIONS:
RESULTS:

Appendix A

STUDENT: STUDENT #6

Mr. Schuman
Teacher

DATE: 7/30/90
SKILL: subtracting, gave strates of
SOFTWARE: Math Masters
OBSERVATIONS: - made several subtraction errors.
- used inflective strategy
- enjoyed enjoyment, Cooperation
RESULTS: Showed a good time

DATE: 7/31
SKILL: Absent
SOFTWARE:
OBSERVATIONS:
RESULTS:

Reference Pedormant test score 2, sign on Computer skill Post comparative

DATE: 8/1
SKILL: Absent
SOFTWARE:
OBSERVATIONS:
RESULTS:

DATE: 8/2
SKILL: Final Testing
SOFTWARE:
OBSERVATIONS:
RESULTS:

N-14 16-0
N-28 24/0
N-30 14/9
N-32 15/0
and
Control skill test P.T. score 12/0

DATE: 8/3
SKILL: Field Day
SOFTWARE:
OBSERVATIONS:
RESULTS:

Appendix B

Target Group Identification Evaluation

ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-mark

N-26
Smaller or Larger (0-10)

3, 10	9, 2	7, 1	4, 8	6, 5	2, 9	10, 7	4, 3	8
7, 9	2, 1	3, 10	8, 6	4, 5	7, 1	9, 10	2, 3	16
5, 4	3, 9	2, 10	6, 4	3, 4	8, 1	10, 2	9, 7	24
7, 10	9, 3	2, 8	5, 1	9, 10	1, 5	2, 3	7, 8	32
9, 1	7, 3	4, 8	5, 2	3, 5	2, 8	9, 4	7, 6	40
5, 8	1, 6	10, 3	6, 2	4, 1	3, 7	9, 5	1, 9	48
6, 7	8, 3	2, 4	5, 9	1, 10	7, 6	7, 9	10, 3	56
2, 7	10, 8	3, 5	6, 9	2, 6	3, 7	3, 1	5, 2	64
1, 8	4, 1	5, 4	8, 6	3, 6	4, 6	6, 10	1, 10	72
2, 3	9, 4	8, 3	4, 8	6, 4	7, 2	3, 2	2, 8	80

Appendix C

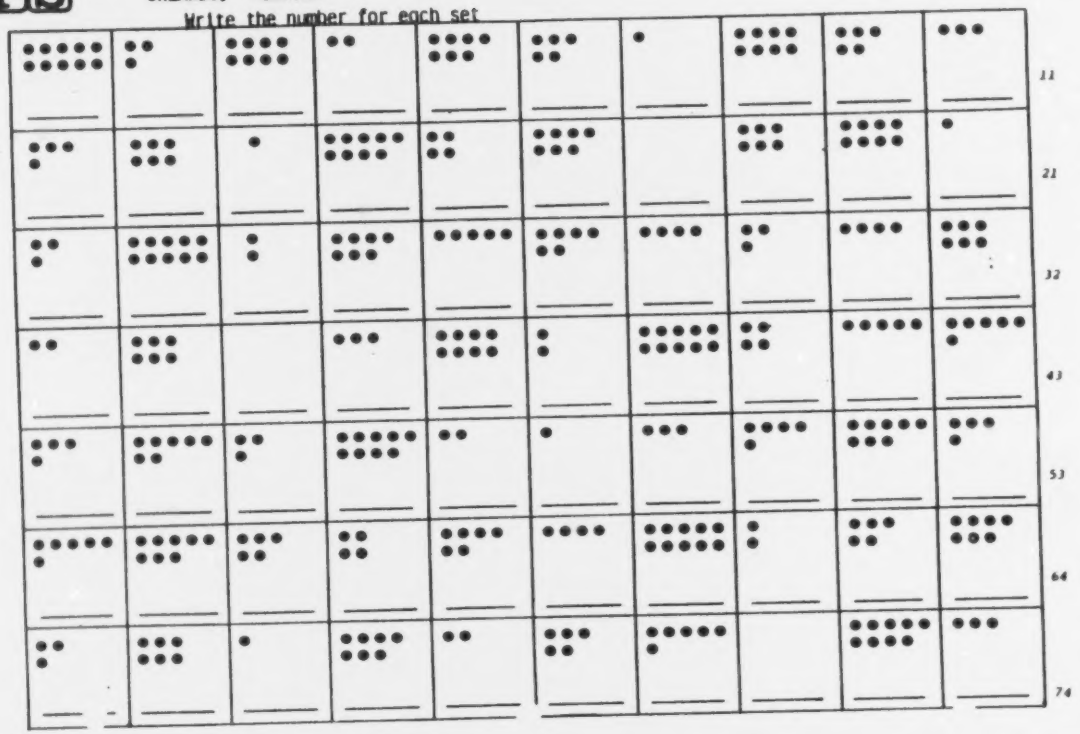
Skill Assessment Pretest

ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-write

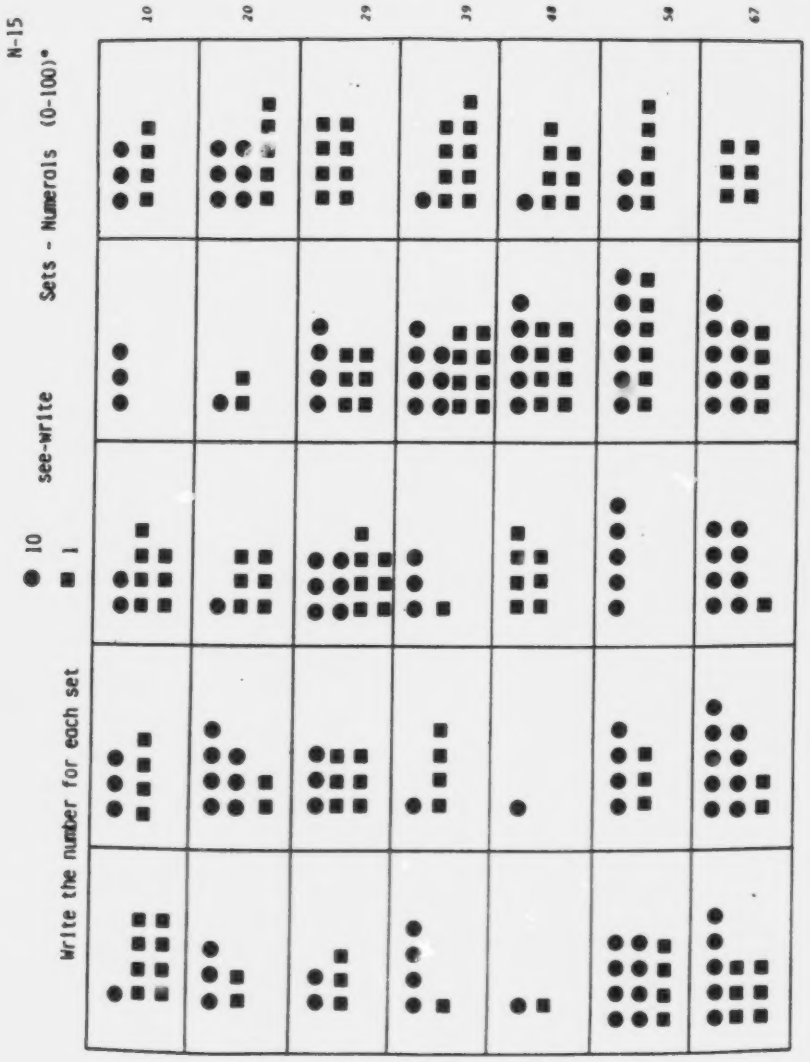
N-14
Sets-Numerals (0-10)

Write the number for each set



Appendix C

Skill Assessment Pretest



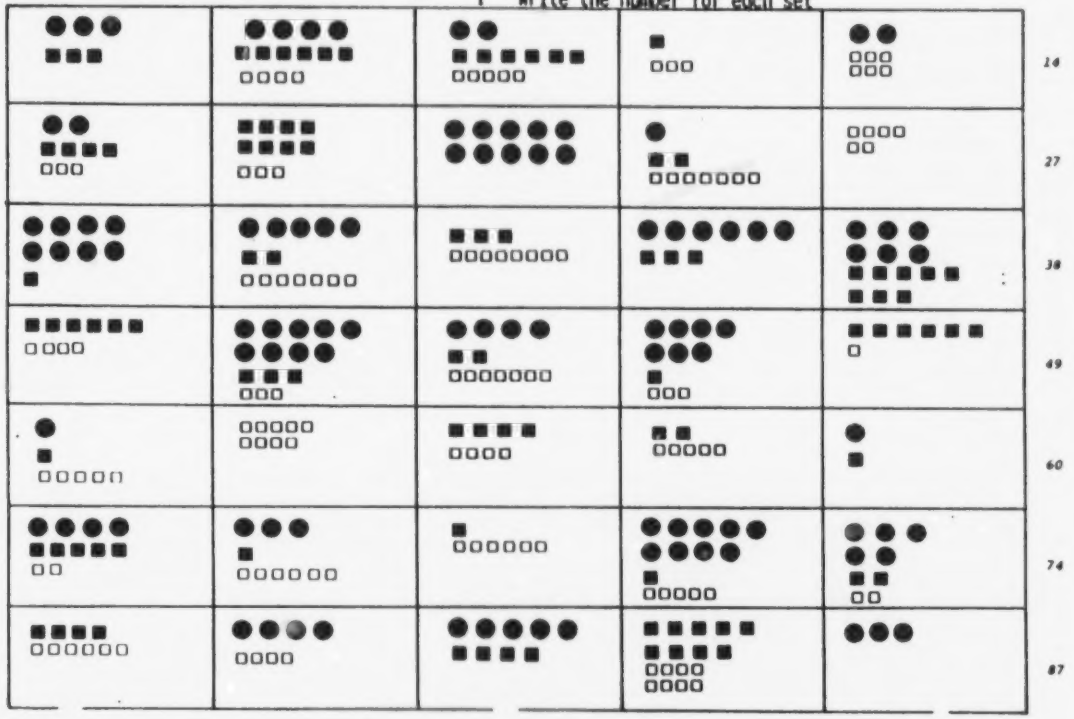
ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

● 100
■ 10
□ 1

see-write

N-16
Sets - Numerals (0-1000)*

Write the number for each set



14
27
38
49
60
74
87

ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-write

N-17

Numerals - Sets (0-10)

Write the dots for each number

3	8	2	7	5	10	4	6	1	9	11
4	7	3	10	2	7	5	6	2	6	22
5	8	1	10	4	7	3	9	2	7	33
4	10	2	8	1	3	9	5	7	6	43
3	6	1	4	7	10	5	8	9	2	53
7	2	4	3	10	8	6	5	1	9	63
10	8	3	5	1	9	2	4	6	7	73

Appendix D

Sign Identification Posttest

Appendix D

Sign Identification Posttest

NAME:_____ DATE:_____ SIGN IDENTIFICATION EVALUATION

CHOICES: LESS THAN, GREATER THAN, EQUAL TO

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.

Appendix E

Student Preference Survey

Appendix E

Student Preference Survey

NAME:_____ DATE:_____ INSTRUCTIONAL PREFERENCE

DIRECTIONS: List in order from 1 to 5 (most liked to least liked) the kind of classroom activities that you could find yourself doing.

_____ learn by working with a story in a book

_____learn by working on a blackboard

_____learn by working on a sheet of paper

_____learn by working with computer software

_____learn by working with a stepboard

Appendix F

Computer Practicum Contract

Appendix F

Computer Practicum Contract

Contingency Contract for Jonathan C. Schuman Summer, 1990

Home Address 5011 S.W. 29 Terrace Phone: 963-5170

Ft. Lauderdale, Fl. 33312 SS# 051-48-2064

The contingencies listed below are considered essential for developing or strengthening the skills which are this practicum's objectives. Section A lists contingencies which are expected of all practicum students, please read each one carefully. Section B lists specific contingencies which are individualized for the particular practicum student. Section C lists the commitments made by the practicum advisor.

A. GENERAL CONTINGENCIES

I, Jonathan C. Schuman, understand that in partial fulfillment of my 6 or 9 credit hour practicum toward my Masters degree in Computer Science Ed., I will .....

1. Attend all required orientation meetings before the summer program begins.
2. Teach in the Super Marks computer summer program from 9:00 a.m. to 12:00 p.m. on Monday through Friday from June 4, 1990 through August 10, 1990.
3. Assist in setting up the computer lab to enhance precision teaching.
4. Demonstrate my knowledge of software, teaching strategies, and evaluation procedures.
5. Gather software to supplement what is already available.
6. Prepare a written inventory list at the end of the summer of the available hardware and software in the computer lab.
7. Preview software and recommend those that should be acquired by the Super Marks program.
8. Gather from teachers and parents intake information about the children assigned to me.
9. Combine instructional efforts with the regular Super Marks teachers.
10. Plan appropriate computer activities for each student assigned to me.
11. Confer with the other computer teachers.
12. Keep a record of the software used and the progress made by each student assigned to me.
13. Keep written daily logs of my activities in the computer lab.
14. Conduct at least one parent conference to share the student's progress.
15. Give parents a written list of appropriate follow-through software that would be beneficial to the child.
16. Maintain open communication with my practicum advisor and discuss problems as the need arises.

Appendix F

Computer Practicum Contract

17. Contact my advisor, as needed, in person or by phone during breaks or after school.
18. Implement suggestions made by my advisor.
19. Submit a computer search of the literature on my practicum topic.
20. Write a final report to share my summer computer lab experience with the educational community.

B. SPECIFIC INDIVIDUAL CONTINGENCIES

1.
2.
3.
4.

5/10/90

C. COMMITMENTS BY THE PRACTICUM ADVISOR

1. Design group seminars to demonstrate teaching strategies and management techniques. Provide an opportunity to share successes and solve problems.
2. Maintain individual written records of each practicum student's work and progress toward meeting the requirements of this contract.
3. Observe the practicum student teaching and provide written feedback with suggestions.
4. Arrange time for conferences on an as needed basis as indicated by the practicum teacher.
5. Encourage communication by phone or in person during breaks or after school.
6. Provide samples of work as models for clinical teaching.

This contract was read and explained to me on 5/10/90

I reviewed this contract and agree to the contents described herein.

DATE 5/10/90

Practicum Student Signature

Sheila H. Melin

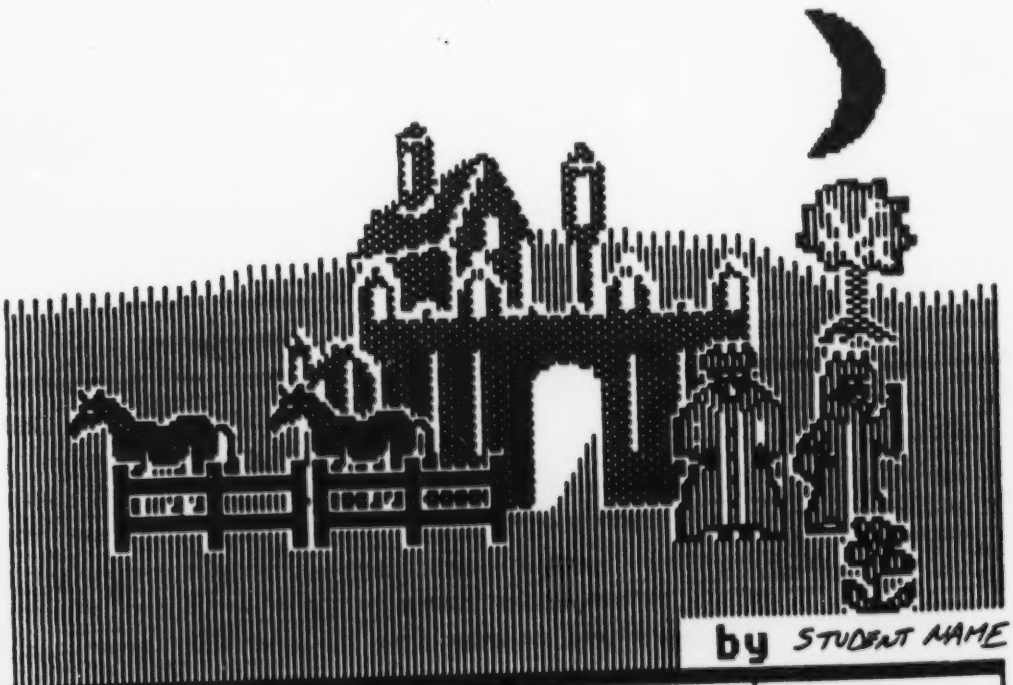
Practicum Advisor Signature

Appendix G

Creative Student Work

Appendix G

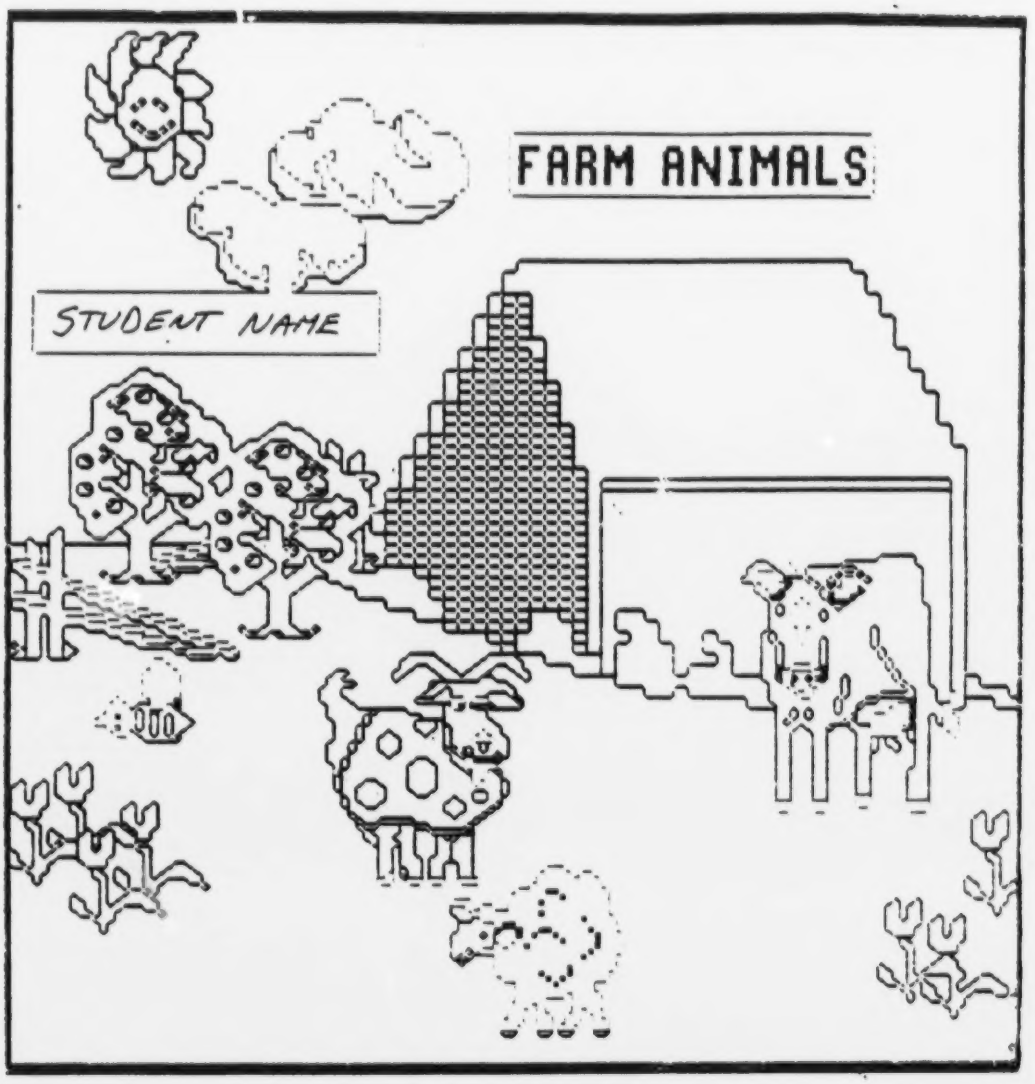
Creative Student Work



fence	flower	gate	horse
king	moon	queen	tree

Appendix G

Creative Student Work



Appendix H

Number Value Comparison Pretest/Posttest

ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-mark

N-26
Smaller or Larger (0-20)*

6, 16	19, 5	12, 1	10, 11	18, 8	4, 3	17, 15	9, 7	1.
14, 2	15, 17	9, 11	7, 13	16, 3	4, 5	12, 18	14, 6	2.
2, 19	8, 10	11, 1	12, 6	2, 9	15, 10	4, 8	19, 18	3.
3, 16	5, 17	7, 10	17, 9	1, 5	4, 14	3, 6	16, 8	4.
18, 15	19, 12	11, 2	9, 7	2, 12	6, 19	18, 15	17, 1	5.
11, 3	8, 5	10, 4	16, 14	6, 9	19, 7	7, 12	15, 14	6.
3, 2	16, 4	18, 11	5, 10	8, 1	14, 8	4, 3	11, 13	7.
6, 17	10, 5	2, 7	18, 16	12, 19	9, 15	12, 6	19, 10	8.
1, 18	14, 11	4, 5	7, 17	3, 15	16, 2	9, 8	1, 8	9.
5, 19	6, 15	2, 10	12, 11	7, 4	3, 14	17, 9	16, 18	10.

Number Value Comparison Pretest/Posttest

ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-mark

N-30
Smaller or Larger (0-50)

0, 50	45, 7	37, 27	19, 33	35, 6	23, 24	5, 4	21, 22	8
16, 15	7, 18	36, 47	40, 26	12, 11	29, 41	46, 13	25, 30	16
1, 3	44, 2	43, 50	3, 2	34, 10	25, 24	10, 19	20, 47	24
41, 43	2, 38	27, 25	14, 11	27, 19	34, 36	0, 1	15, 16	32
40, 26	17, 18	36, 47	29, 41	10, 32	9, 4	13, 42	14, 39	40
8, 46	45, 7	0, 50	13, 12	19, 33	16, 12	28, 30	3, 9	48
37, 27	14, 26	42, 38	7, 18	31, 33	39, 2	50, 0	47, 2	56
35, 4	21, 36	9, 8	49, 48	40, 36	44, 50	41, 39	17, 23	64
32, 11	27, 24	2, 38	3, 5	43, 8	15, 21	8, 9	3, 19	72
29, 30	5, 16	3, 20	18, 24	44, 13	39, 14	19, 10	12, 37	80

OCPS ORANGE COUNTY PUBLIC SCHOOLS
PRECISION TEACHING PROJECT
ORLANDO, FLORIDA

see-mark

N-32
Smaller or Larger (0-100)

35, 6	21, 22	9, 8	67, 72	36, 47	76, 67	23, 24	1, 3	8
44, 2	92, 84	0, 50	14, 11	19, 33	16, 15	12, 11	28, 30	16
5, 8	83, 57	43, 50	63, 26	55, 15	34, 10	82, 28	25, 24	24
16, 61	58, 56	37, 27	24, 36	62, 59	27, 29	5, 4	25, 26	32
13, 10	76, 85	62, 78	40, 26	32, 96	12, 9	29, 41	64, 48	40
91, 92	46, 82	74, 47	15, 12	27, 19	34, 36	0, 1	48, 84	48
54, 45	19, 91	22, 90	38, 49	98, 97	78, 87	6, 2	98, 89	56
68, 52	84, 96	88, 78	3, 2	37, 73	31, 33	98, 100	57, 55	64
49, 50	42, 41	79, 80	65, 96	7, 9	18, 24	72, 50	87, 19	72
99, 28	100, 99	8, 9	69, 68	55, 68	70, 66	90, 96	1, 3	80

Appendix I

Comparing Simple Open Mathematical Sentences
Pretest/Posttest

Appendix I

Comparing Simple Open Mathematical Sentences
Pretest/Posttest

NAME:

DATE:

CHOICES: < , = , >

6+3 5+5	0+1 3+0	4+1 3+6	5+2 3+4	2+2 0+5
8+5 7+7	4+8 7+6	3+6 4+4	0+5 3+1	12+8 9+9
11+9 10+7	5+14 12+8	11+7 8+8	0+1 1+0	5+4 1+10
2+0 0+0	1+0 0+2	4+1 3+3	14-6 12-2	7-4 12-9
15-5 18-9	12-5 20-13	13-8 6-1	20-4 16-2	15-1 13-0
12-5 16-8	16-11 5-1	20-1 18-0	19-3 20-3	23-2 21-1
19-3 20-4	0-0 9-8	17-9 15-4	12-12 5-5	5-4 16-14
11-5 18-12	12-1 8+2	13-4 3+0	15-5 5+4	17-8 5+2
8-4 3+2	19-5 8+6	20-4 3+10	9+2 20-12	2+2 18-14
6+10 20-2	9+9 20-3	22-3 12+8	5+9 19-3	11-11 19-18

BEST COPY AVAILABLE

Appendix J

Instructional Preference Survey

Appendix J

Instructional Preference Survey

NAME:_____ DATE:_____

DIRECTIONS: List in order from 1 to 5 (most liked to least liked) the kind of classroom activities that you could find yourself doing.

_____ learn by working with a story in a book

_____learn by working on a blackboard

_____learn by working on a sheet of paper

_____learn by working with computer software

_____learn by working with a stepboard

Appendix K

Software Recommendation Sheet

Appendix K

Software Recommendation Sheet

The following software programs have proven to be useful in helping your child develop skills in mathematics and language arts. The software packages listed are for use on Apple computers.

MATHEMATICS	LANGUAGE ARTS
Skill: _____	Skill: _____
_____	_____
_____	_____
Software: _____	Software: _____
_____	_____
_____	_____
Skill: _____	Skill: _____
_____	_____
_____	_____
Software: _____	Software: _____
_____	_____
_____	_____

BEST COPY AVAILABLE